

## **Varavoima-asemien etäohjaus ja -valvonta**

Olli Ulmala

Opinnäytetyö

Toukokuu 2018

Tekniikan ja liikenteen ala

Insinööri (AMK), energiatekniikan tutkinto-ohjelma

Tekijä(t) Ulmala, Olli	Julkaisun laji Opinnäytetyö, AMK	Päivämäärä Toukokuu 2018
	Sivumäärä 57	Julkaisun kieli Suomi
		Verkkojulkaisulupa myönnetty: x
Työn nimi <b>Varavoima-asemien etäohjaus ja -valvonta</b>		
Tutkinto-ohjelma Insinööri (AMK), Energiatekniikan koulutusohjelma		
Työn ohjaaja(t) Kari Hytönen, Ari Kuisma		
Toimeksiantaja(t) Finno Energy Oy		
<p>Tiivistelmä</p> <p>Mobiiliverkkojen toimiminen on nykypäivän yhteiskunnalle tärkeää ja tulevaisuudessa se tulee olemaan vielä suuremmassa roolissa. Tukiasemat ovat liitettynä sähköverkkoon, joten niiden toiminta ei ole taattua pitkien sähkökatkojen aikana. Toimeksiantajayrityksellä oli tarve kehittää etäohjattava ja -valvottava varavoimajärjestelmä pilotointikohdetta varten. Tavoitteena oli toteuttaa järjestelmän prototyyppi, joka toimii täysin itsenäisesti sähkökatkojen aikana ja sen tuli lähettää ilmoituksia mahdollisista sähkökatkoista ja käynnistymisistään. Järjestelmän tuli sisältää tarvittavat mittaukset ja niiden mittaustulosten tuli olla saatavilla jatkuvasti.</p> <p>Työ oli tuotekehitysprojekti, jonka kehittämiskohteena oli etäohjaus ja -valvontajärjestelmä. Tietoperusta kerättiin pääsääntöisesti kirjallisuudesta sekä haastatteleamalla toimeksiantajayritystä järjestelmän ominaisuuksista. Järjestelmää lähdettiin kehittämään toimeksiantajayrityksen toiveiden mukaisesti.</p> <p>Teknisen toteutuksen alustana toimi Raspberry Pi 3- tietokone. Siihen liitettiin kaikki mittaukseen ja ohjaukseen tarvittavat komponentit. Etäohjaus- ja valvontaohjelma ohjelmointiin Rasperry Pi:lle ja ohjelman käyttöliittymää päästiin käyttämään toiselta laitteelta VNC-yhteyden avulla. Käyttöliittymän avulla varavoimakoneella voitiin suorittaa myös manuaalisia testiajoja.</p> <p>Lopputuloksena oli toimeksiantajayrityksen tarpeiden mukainen järjestelmä, joka toimitettiin testikäyttöön. Järjestelmää ja sen ominaisuuksia testattiin pilotointikohteessa simuloimalla sähkökatkoja. Konkreettisen lopputuloksen lisäksi saatiin myös tietoa, kuinka järjestelmän seuraavasta versiosta voitaisiin tehdä vielä parempi.</p>		
Avainsanat ( <a href="#">asiasanat</a> ) varavoima, Rasperry Pi, etäohjaus, etävalvonta		
Muut tiedot ( <a href="#">salassa pidettävät liitteet</a> )		

Author(s) Ulmala, Olli	Type of publication Bachelor's thesis	Date May 2018
		Language of publication: Finnish
	Number of pages 57	Permission for web publication: x
Title of publication <b>Remote control and monitoring of backup power sites</b>		
Degree programme Degree Programme in Energy Technology		
Supervisor(s) Kari Hytönen, Ari Kuisma		
Assigned by Finno Energy Oy Ltd		
<p>Abstract</p> <p>The functionality of cellular network is important for the modern society and, in the future, it will be even more important. Cellular sites are connected to an electrical grid, which is why their functionality cannot be guaranteed during long power outages. The assignor wanted to develop a backup power system that can be controlled and monitored remotely. The main objective was to develop a working prototype of a system which is capable of working on its own during power outages and can send notifications about power outages and its startups. The system needed to contain all the relevant measurements and the data of the measurements needed to be accessible all the time.</p> <p>The job was a product development project. The theoretical basis was mainly gathered from literature and by interviewing the assignor's employees to get information on the kind of features the system should have.</p> <p>The platform for the technical solution was a Raspberry Pi 3- computer. All the required measurement and control electronics were connected to it. The remote control and monitoring program was programmed on it and the graphical user interface could be accessed from another computer by VNC -connection.</p> <p>The result of the work was a system that fulfilled the assignor's functional requirements that was delivered for test use. The system was tested at the piloting site by simulating power outages and the system worked as planned. In addition to the physical result, some follow-up actions were determined to make the next version of the system better.</p>		
Keywords/tags ( <a href="#">subjects</a> ) backup power, Raspberry Pi, remote control, remote monitoring		
Miscellaneous ( <a href="#">Confidential information</a> )		

## Sisältö

<b>1</b>	<b>Johdanto .....</b>	<b>5</b>
1.1	Opinnäytetyön tausta ja toimeksiantaja .....	5
1.2	Opinnäytetyön toteutus, tavoitteet ja rajaus .....	5
<b>2</b>	<b>Sähköverkon häiriöt ja varavoima .....</b>	<b>6</b>
2.1	Sähköverkon häiriötilanteet .....	6
2.1.1	Häiriöiden syyt .....	6
2.1.2	Pika- ja aikajälleenkytkentä .....	7
2.1.3	Häiriöiden aiheuttamat ongelmat televerkolle .....	8
2.2	Varavoima .....	9
2.3	Teholähteet .....	9
2.3.1	UPS-laitteet .....	9
2.3.2	Polttomoottorigeneraattorilaitokset .....	10
2.3.3	Akustot .....	11
<b>3</b>	<b>Mittaus- ja ohjausmenetelmät .....</b>	<b>11</b>
3.1	Ohjaustekniikat .....	11
3.1.1	Rele .....	11
3.1.2	Transistori .....	12
3.2	Mittausmenetelmät .....	13
3.2.1	Induktiivinen virranmittaus .....	13
3.2.2	Ultraäänianturi .....	14
3.2.3	Termistori .....	14
3.2.4	Kapasitiivinen kosteusanturi .....	15
3.2.5	A/D-muunnin .....	15
<b>4</b>	<b>Raspberry Pi-tietokone .....</b>	<b>16</b>
4.1	Raspberry Pi 3 Model B-tietokone .....	16

4.1.1	Tekniset tiedot .....	17
4.1.2	Raspbian käyttöjärjestelmä .....	18
4.2	Python-ohjelmointikieli .....	18
4.3	VNC-protokolla ja RealVNC-ohjelma .....	19
<b>5</b>	<b>Varmennettavan kohteen ja varavoimajärjestelmän kuvaus .....</b>	<b>20</b>
5.1	Varmennettavan kohteen kuvaus .....	20
5.2	Varavoimakone.....	20
5.3	Virransyöttö.....	21
<b>6</b>	<b>Tarvittavien mittausten ja ohjausten suunnittelu sekä toteutus .....</b>	<b>22</b>
6.1	LTspice-simulointiohjelma.....	22
6.2	Virta-anturit.....	22
6.2.1	Käytettävät virta-anturit.....	22
6.2.2	Virta-anturien signaalien tasasuuntaus.....	24
6.3	Polttoaineen määrän mittaus .....	26
6.4	Käynnistysakun jännitteen mittaaminen .....	28
6.5	Varavoimakoneen kotelon sisälämpötilan mittaaminen .....	29
6.6	A/D-muunnin.....	31
6.7	Releet.....	32
<b>7</b>	<b>Etäohjaus- ja valvontaohjelmiston ohjelmoiminen .....</b>	<b>34</b>
7.1	Automaattisen ohjelman toimintaperiaate .....	34
7.1.1	Pääohjelma .....	34
7.1.2	Automaattinen käynnistys.....	35
7.1.3	Käyntiohjelma .....	37
7.2	Käyntitietojen tallentaminen .....	38
7.3	Ohjelman käyttöliittymä ja sen hallitseminen .....	39

<b>8 Tulokset .....</b>	<b>40</b>
<b>9 Pohdinta ja jatkotoimenpiteet .....</b>	<b>42</b>
<b>Lähteet .....</b>	<b>44</b>
<b>Liitteet .....</b>	<b>47</b>
Liite 1. Raspberry Pi Model 3 B'n liitännät (Raspberry Pi Pinout Diagram   Circuit Notes. N.d.) .....	47
Liite 2. Etäohjauksen kytkentä (Generator EU70is Owner's Manual. N.d. muokattu) 48	
Liite 3. Generaattorin syötön virranmittauksen mittausjärjestely .....	49
Liite 4. Käynnistysakun jännitemittauksen mittausjärjestely .....	50
Liite 5. Käynnistysakun, lämpötilan, kosteuden ja polttoaineen mittausten koodi 51	
Liite 6. Pää- ja käyntiohjelman koodi .....	52
Liite 7. Varavoimakoneen automaattisen käynnistyksen koodi .....	53
Liite 8. Käyttöliittymän mittausten koodi .....	54
Liite 9. Testiajon aikana tulleet sähköposti-imoitukset .....	55
Liite 10. Testiajon aikana tallennetut mittaustiedot .....	56
Liite 11. Prototyypin lopulliset kytkennät.....	57

## Kuviot

Kuvio 1. Keskeytysmäärien osuudet aiheuttajittain .....	7
Kuvio 2. Keskeytysmäärän jakauma .....	8
Kuvio 3. Releen rakenne ja toimintaperiaate.....	12
Kuvio 4. Npn-transistorin rakenne ja symboli .....	13
Kuvio 5. Sähkömagneettiseen induktioon perustuvan virranmittauksen periaatekuva .....	14
Kuvio 6. Raspberry Pi 3 Model B-tietokone .....	17
Kuvio 7. VNC:n toimintaperiaate.....	19
Kuvio 8. Varavoimajärjestelmän tekninen piirustus .....	21
Kuvio 9. Talema AC1020-virta-anturi .....	23
Kuvio 10. CCT323047-virta-anturi .....	23
Kuvio 11. Vaihtovirran kokoaaltosuuntaus .....	24
Kuvio 12. LTspice-ohjelmalla tehty kuva tukiaseman virransyötön mittausjärjestelystä .....	25
Kuvio 13. Kuvakaappaus LTspice-ohjelmalla tehdystä mittausjärjestelyn simulaatiosta .....	25
Kuvio 14. HC-SR04 ultraäänianturi .....	26
Kuvio 15. HC-SR04 timing diagram.....	27
Kuvio 16. Käynnistysakun jännitemittauksen simuloinnin tulos .....	29
Kuvio 18. Lämpötila- ja kosteusanturi DHT22 (Digikey n.d.).....	30
Kuvio 19. Generaattorin syöttöä mittaavan anturin palauttama signaali virrasta riippuen .....	31
Kuvio 20. Releen ohjaus transistorin avulla .....	33
Kuvio 21. Pääohjelman vuokaavio .....	35
Kuvio 22. Automaattisen käynnistykseen vuokaavio .....	37
Kuvio 23. Käyntiohjelman vuokaavio .....	38
Kuvio 24. Funktio, jolla ohjelma tallentaa mitattavia suureita CSV-tiedostoon.....	39
Kuvio 25. Etäohjaus- ja valvontaohjelman käyttöliittymä tilanteessa, jossa varavoimakone ei ole käynnissä.....	40
Kuvio 26. Työn tuloksena syntynyt prototyyppi .....	41

# 1 Johdanto

## 1.1 Opinnäytetyön tausta ja toimeksiantaja

Infrastruktuurit ovat nykyaikaisen yhteiskunnan perusta ja kaksi tärkeintä niistä ovat tele- ja sähköverkko. Jos näissä verkoissa esiintyy vakavia häiriöitä tai ne eivät ole käytettävissä niin yhteiskunnan toimivuus ei ole taattua. Samalla ihmiset pitävät itsestään selvyytenä sitä, että mobiiliverkko on käytettävissä missä vain ja milloin vain. Mobiililiittymiä on globaalisti enemmän, mitä maapallolla on ihmisiä ja liittymien lukumäärä tulee kasvamaan entisestään, koska erilaiset IoT-sovellutukset tulevat yleisymmään. Yhteiskunta tulee tulevaisuudessa olemaan entistä enemmän riippuvaisempi mobiiliverkon toiminnasta. (Säe 2017, i.)

Toimeksiantajayritys oli vuonna 2013 perustettu espoolainen startup-yritys, joka työllistää tällä hetkellä viisi henkilöä. Yritys kehittää päätoimenaan painetta nostavaa polttoprosessia, mutta tarjoaa myös erilaisia varavoimaratkaisuja. Toimeksiantajayritys halusi lähteä kehittämään erityisesti mobiilitukiasemille tarkoitettua etäohjattavaa ja -valvottavaa varavoimajärjestelmää. Koko varavoimajärjestelmästä oli tarkoitus tehdä mahdollisimman kustannustehokas, koska varavoimakoneen käyttötunnit jäivät yleensä hyvin mataliksi, jolloin yrityksillä ei välttämättä ole suurta halua investoida kalliisiin järjestelmiin. Vaikka käyttötunnit jäivät mataliksi niin toimivat varavoimajärjestelmät ovat hyvin arvokkaita pitkien sähkökatkojen ajalla. Varavoimakoneen etäohjaus- ja -valvontaominaisuudella oli tarkoitus saada lisättyä järjestelmän käyttövarmuutta.

## 1.2 Opinnäytetyön toteutus, tavoitteet ja rajaus

Opinnäytetyö toteutettiin tuotekehitysprojektina, jonka tavoitteena oli kehittää prototyyppilaitteisto, jolla on mahdollista ohjata ja valvoa varavoimajärjestelmää etänä. Fyysisen prototyypin lisäksi tavoitteena oli myös saada jatkokehitysideoita, kuinka prototyypistä voitaisiin tuottaa valmis tuote. Toimivan prototyypin tuli olla valmis opinnäytetyön loppupuolella, jolloin koko järjestelmän oli suunniteltu toimitettavaksi



pilotointikohteeseen testausta varten. Työn tuloksia oli tarkoitus analysoida testamalla järjestelmää pilotointikohteessa, jolloin saatiin tietoa, toteuttiko prototyyppi sille alussa asetetut tavoitteet.

Toimeksiantajayritys määrittäi työn alussa mitä ominaisuuksia he prototyypiltä toivoivat, mutta antoivat vapaat kädet toteutustapojen ja -tekniikoiden suhteen. Tiivistettynä laitteiston tuli toimia täysin itsenäisesti sähkökatkojen aikana ja lähettää ilmoituksia käyttäjälle toiminnoistaan. Etäohjaus- ja -valvontalaitteen prototyypin tuli sisältää seuraavat ominaisuudet

- varavoimakoneen automaattinen käynnistäminen sähkökatkojen ajaksi
- ilmoitukset käyttäjälle varavoimakoneen toiminnoista sähkökatkojen ajalta
- mahdollisuus suorittaa manuaalisia testiajoja etänä
- eri mittauksia
  - sähköverkosta tukiasemalle syötettävä virta
  - generaattorin syöttämä teho
  - jäljellä oleva polttoaine
  - varavoimakoneen kotelon sisäinen lämpötila
  - varavoimakoneen käynnistysakun jännite
- käyttöliittymä, josta on mahdollista lukea mitattavat suureet sekä suorittaa manuaaliset testaukset

Opinnäytetyö rajattiin koskemaan vain kehitettävää prototyyppilaitteistoa, joten työssä ei ollut tarkoituskaan kehittää laitetta, joka soveltuisi sellaisenaan laajemmassa mittakaavassa käytettäväksi.

## 2 Sähköverkon häiriöt ja varavoima

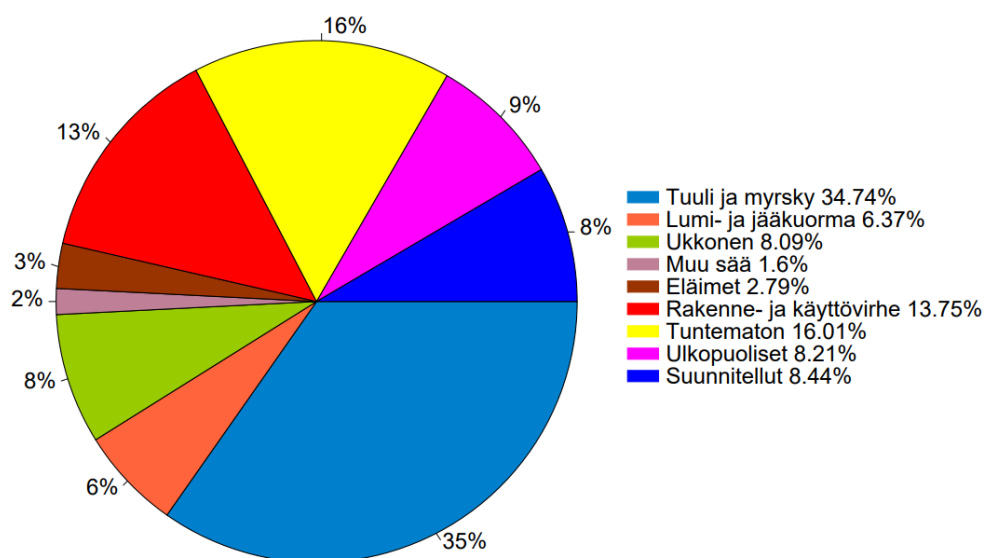
### 2.1 Sähköverkon häiriötilanteet

#### 2.1.1 Häiriöiden syyt

Suomalainen sähköverkko toimii luotettavasti, mutta joskus siinäkin esiintyy eri syistä johtuvia häiriöitä. Sähköverkon häiriötilanteita aiheuttavat tekniset viat ja erilaiset luonnonilmiöt. Luonnon aiheuttamat riskit ovat salamat, lumi, jää, tulvat, kaatuvat puut sekä kova pakkanen. Teknisten vikojen aiheuttamat häiriöt eivät yleensä kestä pitkään. Teknisten vikojen aiheuttamat häiriöt eivät myöskään yleensä leviä laajalle alueelle, koska verkko on suunniteltu siten, että vika-alue voidaan eristää muusta

verkosta. Luonnonilmiöt vaikuttavat myös katkojen pituuteen oleellisesti, esimerkiksi pitkään kestävä myrsky todennäköisesti pitkittää sähkökatkoja ja vaikeuttaa teknisten vikojen korjaamista. (Yleistietoa häiriöistä n.d.)

Kuviossa 1 on esitetty mistä syystä vuoden 2016 sähkökatkot johtuivat.



Kuvio 1. Keskeytysmäärien osuudet aiheuttajittain (Keskeytystilasto 2016, 2017, 4.)

### 2.1.2 Pika- ja aikajälleenkytkentä

Epäonnistunut jälleenkytkentä on lyhyin sähköverkon katkos, joka tilastollisesti tulkitaan katkoksi. Jälleenkytkentöjä on kahta eri tyyppiä, pikajälleenkytkentä (PJK) ja aikajälleenkytkentä (AJK). PJK:lla tarkoitetaan johtoa syöttävän katkaisijan avautumista vian seurauksena ja sen automaattista sulkeutumista noin puolen sekunnin jälkeen. Jos vika on hävinnyt tuon puolen sekunnin aikana, niin sähkönjakelu jatkuu normaalisti jälleenkytkennän jälkeen. Jos vika ei kuitenkaan korjaannu PJK:lla, niin sen jälkeen suoritetaan AJK, joka on toimintaperiaatteeltaan samanlainen kuin PJK, mutta siinä katkaisijat pysyvät auki noin 1-3 minuutin verran. (Varmennetut sähkönjakelujärjestelmät 2005, 14.)

Kuviossa 2 on esitetty, kuinka suuri osa vuonna 2016 tapahtuneista verkon vikatilanteista selvitettiin PJK:lla ja AJK:lla sekä kuinka suuri osa jäi pitkiksi keskeytyksiksi.



Kuvio 2. Keskeytysmäärän jakauma (Keskeytystilasto 2016, 2017, 4.)

Esimerkiksi ukkosilmalla johtimeen iskevä salama synnyttää valokaaren johtimen ja pylvään välissä olevan eristimen yli. Johtimessa vaikuttava jännite ylläpitää valokaarta eikä sitä saada sammumaan muuten kuin kytkemällä johdin jännitteettömäksi. Johtimesta voidaan kytkeä jännite pois suojareleiden ja katkaisijoiden avulla, jolloin valokaari sammuu. Jos valokaari ei kuitenkaan jostain syystä sammuisi, niin PJK:n jälkeen suoritettaisiin AJK, jolloin valokaari saataisiin sammumaan. (Mitä tarkoittavat PJK ja AJK? 2017.)

### 2.1.3 Häiriöiden aiheuttamat ongelmat televerkolle

Nykyään matkapuhelimille ei ole vaihtoehtoa varsinkaan laajoilla ja harvaan asutuilla alueilla, koska kiinteä lankapuhelinverkko on jo suurimmaksi osaksi purettu pois. Tämän vuoksi puhelimen toimivuus ja sähkön saanti ovat entistä enemmän nivoutuneina toisiinsa. Suomessa on kymmeniätuhansia mobiilitukiasemia ja suurinta osaa niistä ei ole varustettu pitkien sähkökatkojen varalle. (Ojala 2018.)

Vuodenvaihteessa 2017-2018 tykkylumi aiheutti Kainuun alueella pitkiä häiriöitä sähköjakelussa, jonka vuoksi myös sähkönsyöttö tukiasemille häiriintyi. Alueen tukiasemat toimivat niille asennettujen akustojen ja varavoiman varassa. Sähköverkon häiriöt olivat kuitenkin niin pitkiä, että osilta tukiasemilta loppui akustojen virta kesken, joka taas aiheutti häiriöitä televerkossa. Tästä johtui, että hätänumeroon soitetessa kaikki puhelut eivät välttämättä menneet läpi hätäkeskukseen asti. (Kainuussa mahdollisia häiriöitä hätänumeroon soitetessa – voi koettaa soittaa ilman sim-korttia 2017.)

## 2.2 Varavoima

Sähkön saanti on nyky-yhteiskunnalle erittäin tärkeää ja melkein koko toiminta pysähtyy, kun sähkön saanti katkeaa. Mitä tärkeämmästä järjestelmästä tai prosessista on kysymys, sitä suuremman haitan pienikin häiriö sähköjakelussa aiheuttaa. Sähkön saatavuutta halutaan siis varmistaa varsinkin tärkeille ja monimutkaisille prosesseille. Varavoiman hankkimisesta aiheutuu tietenkin lisäkustannuksia, jotka riippuvat asennettavasta tehosta, mutta hinta ei kuitenkaan ole suoraan verrannollinen tehoon ja lisäteho on yleensä halvempaa. Tapauksissa, jossa halutaan varmistaa esimerkiksi pari tärkeää laitetta niin voi olla järkevämpää varmistaa koko kohde, jolloin ei tarvitse tehdä erillistä varmistusverkkoa. Varavoimaa hankittaessa on hyvä vertailla erilaisia vaihtoehtoja ja menetelmiä, koska niiden kustannuksissa saattaa olla suuriakin eroja. Vaikka järjestelmän hankintahinta olisi matala niin voi paljastua, että sillä on korkeammat ylläpitokustannukset ja lyhyempi elinkaari verrattuna hankintahinnalta korkeampaan ratkaisuun. (Varmennetut sähköjakelujärjestelmät 2005, 13.)

## 2.3 Teholähteet

Sähkön saanti voidaan varmistaa hyödyntämällä erilaisia tekniikoita kuten UPS-laitteita, polttomoottorigeneraattorilaitoksia sekä akustoja.

### 2.3.1 UPS-laitteet

UPS (Uninterruptible Power System) tarkoittaa keskeytymättömän tehon järjestelmää. UPS-laitteet voidaan jakaa vielä staattisiin ja dynaamisiin laitteisiin. Staattinen

UPS-laite käyttää tehonsyöttöön sähköverkon virtaa, kun sitä on saatavilla. Sähköverkon häiriötilanteissa järjestelmä käyttää akkuihin varastoitunutta energiaa. Laite pysyy muuttamaan vaihtovirtaa tasavirraksi sekä muuntamaan akustolta tulevan tasavirran vaihtovirraksi. (Varmennetut sähkönjakelujärjestelmät 2005, 59.)

Dynaamisen UPS-laitteen toiminta perustuu pyörivään sähkökoneeseen, jonka käämeiltä saadaan tarvittava sähköenergia. Kun sähköverkko toimii normaalisti niin se syöttää sähkökonetta, joka pyörii. Kun syöttö sähköverkosta katkeaa niin sähkökoneeseen on varastoitunut pyörimisenergiaa, joka voidaan muuttaa sähköenergiaksi. Sähköä riittää yleensä noin millisekuntien ajaksi, tätä aikaa kutsutaan varakäyntiajaksi. Varakäyntiaikaa voidaan pidentää käyttämällä huimamassaan varastoitua pyörimisenergiaa, jonka varakäyntiaika vaihtelee sekunneista minuutteihin. Toinen vaihtoehto varakäyntiajan pidentämiseksi on akuston käyttö. (Varmennetut sähkönjakelujärjestelmät 2005, 67.)

### 2.3.2 Polttomoottorigeneraattorilaitokset

Yleisin käytettävä varavoimakone on dieselmoottorikäyttöinen generaattori. Dieselgeneraattorin pääkomponentit ovat seuraavat:

- dieselmoottori
- generaattori
- moottorin ja generaattorin välinen voimansiirto
- käyntitärinän eristimet
- runko- ja alustarakenne (Varavoimalaitokset 2013, 90.)

Varavoimakoneet voivat toimia joko saarekekäytössä tai sähköverkon rinnalla. Saarekekäytöllä tarkoitetaan tilannetta, jossa varmennettu verkon osa erottautuu omaksi saarekkeeseen sähköverkon häiriön ajaksi. Tällöin varavoimakone syöttää saarekkeeksi erotettua aluetta eikä virtaa pääse saarekkeen ulkopuolelle. Kun syöttöä vaihdetaan sähköverkon ja varavoimakoneen välillä, niin tapahtuu noin 1-2 sekunnin mittainen katko. (Varmennetut sähkönjakelujärjestelmät 2005, 73.)

Kun varavoimakone toimii sähköverkon rinnalla niin se mahdollistaa katkottoman paluun sähköverkkoon eli järjestelmää voidaan syöttää samaan aikaan varavoimakoneesta ja sähköverkosta. Tämä järjestely mahdollistaa myös varavoimakoneen koe-

käytön sähköverkon toiminnan aikana sekä sen, että sillä voidaan tuottaa järjestelmään lisää virtaa, kun kulutus on huipussaan. (Varmennetut sähkönjakelujärjestelmät 2005, 78.)

### 2.3.3 Akustot

Akustoilla voidaan turvata katkeamaton virransyöttö järjestelmille ja laitteille, joiden halutaan toimivan normaalisti myös silloin, kun sähköverkosta ei ole saatavilla sähköä. Tällaisia kohteita ovat muun muassa televerkon tukiasemat ja keskuskeskukset sekä voimalaitosten ja sähköverkon järjestelmät. Akustoja voidaan käyttää myös energian varastointiin, eli niihin voidaan kerätä esimerkiksi aurinkopaneelien päivän aikana tuottama energia ja käyttää se yön aikana, kun aurinko ei paista. Mobiilitukiasemilla käytettäviä akkuja kutsutaan paikallisakuiksi. Nimellä tarkoitetaan akkuja, jotka on tarkoitettu varmistamaan sähkökatkon aikainen virransyöttö ja ne on suunniteltu kestävänsä vuosien ajan jatkuvaa ylläpitovarausta. Akusto tulee sijoittaa sille tarkoitettuihin olosuhteisiin sekä sitä täytyy myös huoltaa, jotta käyttöikä pysyisi mahdollisimman pitkänä. (ST 52.30.01, 2016, 1.)

## 3 Mittaus- ja ohjausmenetelmät

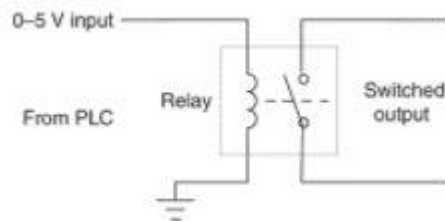
Tässä luvussa käsitellään lyhyesti työssä käytettyjen mittaus- ja ohjausmenetelmien teoriaa ja toimintaperiaatteita.

### 3.1 Ohjaustekniikat

#### 3.1.1 Rele

Releiden avulla voidaan kytkeä suurempia virtoja tai jännitteitä pois tai päälle hyödyntäen pienempää ohjausvirtaa. Kun releen kelan läpi johdetaan virtaa (ohjausvirta), niin se synnyttää ympärilleen magneettikentän, joka vetää releen liikkuvan kärjen kiinni kontaktiin (ks. kuvio 3). Rele voi sisältää useamman kuin yhden kontaktin ja jokainen kontakti suljetaan tai avataan saman kelan avulla. Tällöin yhdellä releellä voidaan ohjata kahta eri virtaa samanaikaisesti. Kontakteja on kahta eri tyyppiä

eli normaalisti auki olevia sekä normaalisti kiinni olevia. Normaalisti auki oleva kontakti sulkeutuu, kun kelaan johdetaan virtaa, kun taas normaalisti kiinni oleva kontakti aukeaa, kun kelaan johdetaan virtaa. (Bolton 2009, 40.)

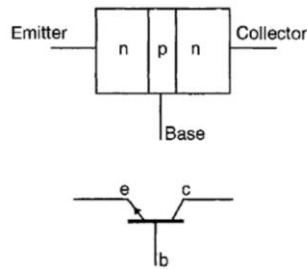


Kuvio 3. Releen rakenne ja toimintaperiaate (Bolton 2009, 40.)

### 3.1.2 Transistori

Transistori on elektroniikassa laajasti käytetty komponentti, jota voidaan käyttää muun muassa kytkimenä ja vahvistimena. Käytettäessä sitä kytkimenä sillä voidaan ohjata suurempia virtoja käyttäen pienempää ohjausjännitettä. (Transistori n.d.)

Transistori koostuu kahdesta tai useammasta n- ja p-tyyppin rajapinnasta. Npn-transistori koostuu hyvin ohuesta kerroksesta p-tyyppin materiaalia, joka on kahden n-tyyppin materiaalista koostuvan kerroksen välissä (ks. kuvio 4). Kuviossa 4 kuvatun transistorin vasemman puolen n-tyyppin materiaali kutsutaan emitteriksi, jolla on pieni negatiivinen jännite suhteessa p-tyyppin materiaaliin, jota kutsutaan kannaksi. Tämän pienen jännite-eron ansiosta virta ei pääse kulkemaan n-p rajapinnan läpi. Kun transistorin kantaan johdetaan virtaa, niin se mahdollistaa virran kulun kollektorilta emitterille. (Warne 2005, 59.)



Kuvio 4. Npn-transistorin rakenne ja symboli (Warne 2005, 59.)

## 3.2 Mittausmenetelmät

### 3.2.1 Induktiivinen virranmittaus

Mitattavassa johtimessa kulkeva vaihtovirta synnyttää ympärilleen muuttuvan magneettivuon. Kun muuttuva magneettivuon lävistää johdinmateriaalista valmistetun silmukan niin silmukkaan indusoituu jännite Faradayn induktiolain mukaan. Yksinkertainen induktiivinen virta-anturi koostuu käämistä (yksi tai useampi kierros), joka on kierretty magneettisen sydämen ympärille sekä mittavastuksesta joka on kytketty käämin päiden välille. Mittavastuksen yli vaikuttava jännite on verrannollinen johtimessa olevaan virtaan. (Iwansson, Sinapius, Hoornaert 1999, 59.)

Kaavassa 1 on esitetty, kuinka käämiin indusoituneen virran suuruus riippuu ensiö- ja toisiopuolen käämien kierroksista.

$$I_2 = \frac{N_1}{N_2} I_1 \quad (1)$$

missä  $N_1$  = ensiöpuolen kierrokset

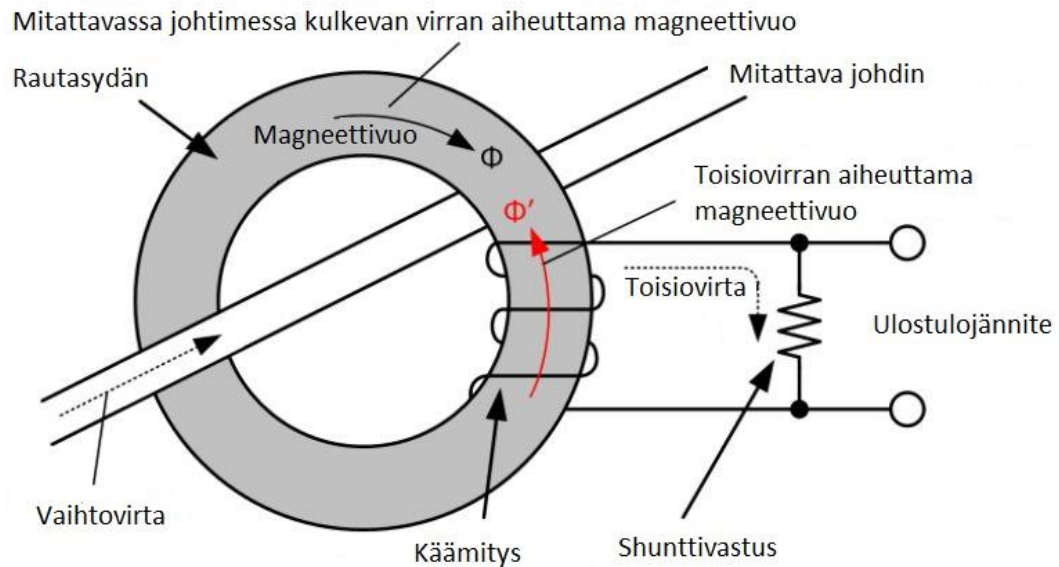
$N_2$  = toisiopuolen kierrokset

$I_1$  = johtimessa kulkeva virta

(Iwansson ym. 1999, 61.)

Kuviossa 5 on esitetty havainnekuva, kuinka induktiivinen virta-anturi toimii.





Kuvio 5. Sähkömagneettiseen induktioon perustuvan virranmittauksen periaatekuva  
(Principle of current sensor n.d. muokattu)

### 3.2.2 Ultraäänianturi

Ultraäänianturit mittaavat kuinka kauan lähetetyllä ultraäänipulssilla kestää kulkea mitattavaan kohteeseen ja takaisin anturin vastaanottimeen. Mitattava kohde voi olla joko kiinteä aine tai nesteen pinta. Kun anturin lähettämä pulssi kohtaa mitattavan kohteen, niin vain pieni osa pulssista lävistää kohteen ja suurin osa siitä heijastuu takaisin anturiin, heijastuvaa osaa kutsutaan kaiuksi. Pelkistettynä anturit koostuvat siis lähettimestä, vastaanottimesta sekä signaalia käsittelevästä elektroniikasta. Anturit käyttävät taajuuksia, joita ihmisen kuuloaisti ei pysty havaitsemaan. Niiden energiankulutus on myös hyvin matalaa, normaalisti muutama watin tuhannesosa. (Hughes 2015, 176.)

### 3.2.3 Termistori

Termistori on elektroniikan komponentti, jonka resistanssi muuttuu nopeasti ympäristön lämpötilan mukaan. Itse termistori on valmistettu puolijohdemateriaalista ja

se on päällystetty jollain eristemateriaalilla, kuten epoksilla. Termistorit voidaan jakaa NTC - ja PTC-termistoreihin. NTC-termistoreiden resistanssi pienenee lämpötilan kasvaessa ja PTC-termistoreiden resistanssi taas kasvaa lämpötilan kasvaessa. Valtaosa käytettävistä termistoreista on NTC-termistoreita. (Gupta, Arora, Westcott 2017, 328-329.)

Termistorit ovat halpa ja luotettava tapa mitata lämpötilaa sovelluksissa, joissa anturin vaaditaan vastaavan nopeasti lämpötilan muutokseen. Termistorin huonoja puolia ovat kapea mitta-alue, ei lineaarinen signaali sekä lämpeneminen, joka vääristää mittaustulosta. (Gupta ym. 2017, 330.)

### 3.2.4 Kapasitiivinen kosteusanturi

Kapasitiivisella kosteusanturilla voidaan mitata ilman suhteellista kosteutta, yleensä välillä 5 – 95 %RH. Anturi koostuu kahdesta elektrodista, jonka välissä on ohuella polymeeri- tai metallioksidikalvolla päällystetty substraatti. Ilmassa oleva vesihöyryä imeytyy substraattiin, kunnes ne ovat tasapainossa eli molempien kosteudet ovat yhtä suuret. Kuivan substraatin suhteellinen permittiivisyys on matalampi, kuin kostean, joten ilmankosteuden muutokset havaitaan anturin kapasitanssin muutoksena. (Cornell 2012, 90.)

### 3.2.5 A/D-muunnin

Nykyaikaiset elektroniset laitteet ovat digitaalisia. Vaikka jotkin lähettimet ja systeemit kommunikoivat keskenään digitaalisessa muodossa, niin esimerkiksi teollisuuden mittauksissa käytettävät anturit ovat analogisia. Jotta digitaaliset laitteet saadaan kommunikoimaan analogisten laitteiden kanssa, tarvitaan analogia-digitaalimuunnos (A/D) ja digitaali-analogiamuunnos (D/A). (Thompson 2006, 299.)

A/D-muunnin muuntaa esimerkiksi anturilta tulevan analogisen jännitesignaalin digitaaliseen muotoon. Tällöin esimerkiksi digitaaliset näytöt tai tietokonepohjaiset tiedonkeräyslaitteet voivat hyödyntää signaalia. A/D muuntimet ovat yleensä sulautettuna sitä käyttävien laitteiden yhteyteen, kuten näyttöihin. A/D-muuntimet esiintyvät harvoin erillisenä muusta mittaussysteemistä. (Cheatle 2006, 130-131.)

Olennaisin A/D-muuntimen ominaisuus on sen resoluutio eli kuinka monta bittiä ulos tuleva signaali sisältää. Yhdellä bitillä voi olla kaksi tilaa, joko 1 tai 0. 8-bittisen A/D-muuntimen antamalla digitaalisella signaalilla voi siis olla  $2^8 = 256$  eri tilaa, kun taas 12-bittisellä voi olla  $2^{12} = 4096$  eri tilaa. Eli mitä suurempi resoluutio sitä enemmän eri jännitetasoja A/D-muunnin kykenee erottamaan. (Cheatle 2006, 131.)

Kaavassa 2 on esitetty kuinka 8-bittisen A/D-muuntimen resoluutio lasketaan (tulevat signaalit välillä 0-10V).

$$\text{Resoluutio} = \frac{10 \text{ V}}{2^8} = 40 \text{ mV} \quad (2)$$

Toinen tärkeä A/D-muuntimen ominaisuus muunnosaika eli kuinka kauan muuntimella kestää muuntaa analogisen signaalin näyte digitaalseksi signaaliksi. Aika perustuu pitkälti sille minkälaisen elektronisen piirin muunnin sisältää. (Cheatle 2006, 131.)

## 4 Raspberry Pi-tietokone

Raspberry Pi on isobritannialaisen Raspberry Pi Foundationin kehittämä yhden piirilevyn tietokone. Se kehitettiin alun perin sitä varten, että yhdistys halusi edistää ihmisten kiinnostumista tietokoneisiin ja tietojenkäsittelytieteisiin (What is a Raspberry Pi? n.d.b). Pienen koon ja matalan hinnan ansiosta myös elektroniikkaharrastajat ovat ottaneet Raspberry Pi'n käyttöön projekteihinsa. (What is a Raspberry Pi? n.d.a.)

### 4.1 Raspberry Pi 3 Model B-tietokone

Raspberry Pi 3 Model B (ks. kuvio 6) on ensimmäinen versio kolmannen sukupolven Raspberry Pi-tietokoneista. Se julkaistiin helmikuussa 2016, jolloin se korvasi edeltäjänsä Raspberry Pi 2 Model B'n. Kolmannen sukupolven uusin malli, Model B+, julkaistiin alkuvuodesta 2018. (Raspberry Pi 3 Model B n.d.)



Kuvio 6. Raspberry Pi 3 Model B-tietokone (Raspberry Pi 3 Model B n.d.)

#### 4.1.1 Tekniset tiedot

Raspberry Pi on täysiverinen Linux-tietokone, joka sisältää kaikki ominaisuudet, joita voidaan odottaa vähän virtaa kuluttavalta tietokoneelta. Siinä on

- Quad Core 1.2GHz Broadcom BCM2837 64bit-prosessori
- 1 Gt keskusmuistia
- BCM43438 WLAN ja Bluetooth
- 4 kappaletta USB 2-porttia
- 3,5 mm:n audioliitin
- HDMI
- CSI-kameraportti
- DSI-portti kosketusnäytölle
- Micro SD-korttipaikka (Raspberry Pi 3 Model B. N.d.)

Raspberry Pi Model 3 B sisältää kaksirivisen pinniriman, joka sisältää 40 kappaletta pinnejä. Neljä niistä toimii jatkuvina virtalähteinä (kaksi 5 VDC:n pinniä ja kaksi 3,3 VDC:n pinniä) eli niitä ei voi ohjelmoida muuhun tarkoitukseen. Virtalähteiden lisäksi pinnirimasta löytyy kahdeksan maapinniä. Loput 26 pinnejä ovat GPIO-pinnejä (General purpose input/output). Jokainen GPIO-pinni voidaan ohjelmoimalla määrittää joko tuloksi (input) tai lähdöksi (output). Lähdöksi ohjelmoidulla pinnillä voi olla kaksi tilaa joko, "high" tai "low". Kun pinnin tila on "high", se antaa 3,3 VDC:n suuruisen

jännitteen, ja kun sen tila on "low", sen jännite on 0 VDC. Tuloksi ohjelmoituihin pinihin voidaan syöttää maksimissaan 3,3 VDC:n jännitettä, jolloin pinnin tilaksi luetaan "high". Kun siihen ei syötetä jännitettä, sen tilaksi luetaan "low". Yksinkertaisten input/output-laitteiden lisäksi GPIO-pinnejä voidaan käyttää moneen muuhunkin tar-koitukseen, kuten

- PWM (pulse-width modulation)
- SPI (tiedonsiirto)
- I2C (tiedonsiirto)
- Serial (tiedonsiirto) (GPIO n.d.)

Liitteessä 1 on esitetty GPIO-pinnien numerointi sekä muiden liitäntöjen paikat.

#### 4.1.2 Raspbian käyttöjärjestelmä

Raspbian on Raspberry Pi'lle suunniteltu ilmainen käyttöjärjestelmä, joka on johdettu Debianista, suositusta Linux-jakelupaketista. Käyttöjärjestelmä koostuu perusohjelmista ja työkaluista, jotka mahdollistavat Raspberry Pi'n käytön, mutta se sisältää myös yli 35 000 esikäännettyä ohjelmistopakettia. Käyttöjärjestelmä on erityisesti optimoitu Raspberry Pi'n laitteistolle. (Welcome to Raspbian n.d.)

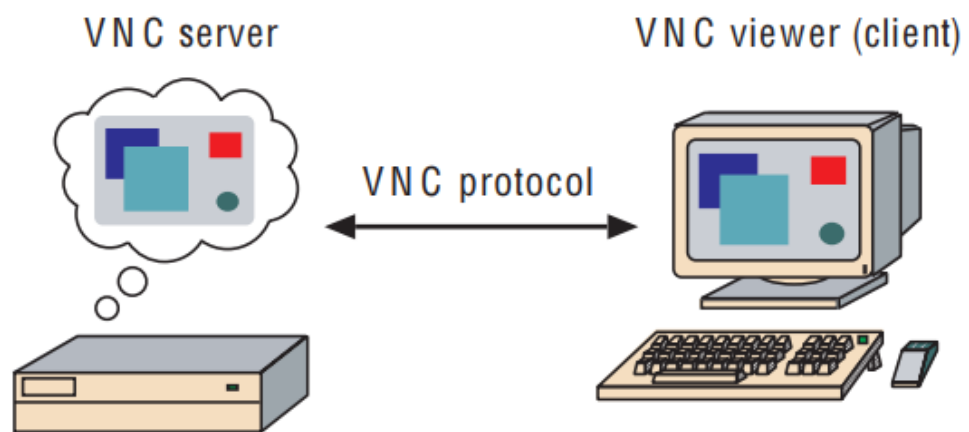
#### 4.2 Python-ohjelmointikieli

Python on tulkattava korkean tason ohjelmointikieli, jonka yksinkertainen syntaksi tekee siitä helposti luettavaa (What is Python? n.d.). Helposti luettavan syntaksin vuoksi sillä on helppo lähteä kirjoittamaan ohjelmia sekä virheiden etsiminen koodista on myös helppoa. Tämän takia sitä pidetään hyvänä ensimmäisenä opeteltavana ohjelmointikielenä. (Rasila 2014.)

Python ohjelmia voidaan tehdä kirjoittamalla koodia tekstieditoriin, jonka jälkeen Python-tulkki lukee kirjoitetun koodin ja kääntää sen tietokoneen ymmärtämään muotoon ja tietokone suorittaa tekstieditoriin kirjoitetut komennot. Tällä tarkoitetaan sitä, että Python on tulkattava kieli. (Opi Pythonia n.d.)

### 4.3 VNC-protokolla ja RealVNC-ohjelma

VNC (Virtual Network Computing) on protokolla, joka kehitettiin vuonna 1995 Cambridgessa Olivetti Research Laboratoryn toimesta. Alkuperäinen koodi on avointa lähdekoodia, joten protokollan sovellutuksesta on olemassa eri yritysten omia versioita. Protokollan avulla voidaan esittää toisen tietokoneen graafinen käyttöliittymä toisella tietokoneella internetin yli (ks. kuvio 7). VNC-ohjelmat ovat erittäin käteviä esimerkiksi etätyöskentelyyn. (Richardson, Stafford-Fraser, Wood, Hopper 1998, 33.)



Kuvio 7. VNC:n toimintaperiaate (Richardson, Stafford-Fraser, Wood, Hopper 1998, 35.)

VNC-palvelimen (VNC server) täytyy olla käynnissä isäntätietokoneella, eli se tietokone jota halutaan hallita etänä internetin yli. VNC-pääteohjelma (VNC viewer) tulee taas olla asennettuna sille tietokoneelle, jolta halutaan hallita isäntätietokonetta. Isäntä- ja asiakastietokoneet tulee olla yhdistettynä samaan TCP/IP verkkoon, joka voi olla lähiverkko, VPN tai julkinen verkko kuten internet. Palvelintietokoneeseen voidaan muodostaa yhteys useammalta asiakastietokoneelta. Yhteyden muodostaminen onnistuu myös esimerkiksi älypuhelimesta, johon on asennettu VNC pääteohjelma. (VNC User Guide 2015, 10.)

Tietokoneiden välinen yhteys on mahdollista salata. Salauksella varmistetaan, että mikään kolmas osapuoli ei pääse käsiksi kahden tietokoneen väliseen tiedonsiirtoon. Yhteys salataan joko 128- tai 256-bittisellä AES (Advanced Encryption Standard) lohkosalausmenetelmällä. (VNC User Guide 2015, 22.)

Raspbian käyttöjärjestelmän mukana tulee valmiiksi asennettu RealVNC'n VNC Connect -ohjelma, joka sisältää palvelin- ja pääteohjelman. Palvelinohjelma täytyy vain aktivoida niin se on valmiina käytettäväksi. (VNC (Virtual Network Computing) n.d.)

## **5 Varmennettavan kohteen ja varavoimajärjestelmän kuvaus**

### **5.1 Varmennettavan kohteen kuvaus**

Valtaosa varmennettavan tukiaseman laitteista toimivat 48 V tasavirralla, pois lukien laitesuojan valaistus sekä ilmastoinnin puhaltimet. Laitteiden keskimääräinen virrankulutus oli noin 80 A eli hieman vajaa 4 kW. Tukiaseman maksimi tehontarve oli hieman yli 5 kW.

Tukiasemalle oli asennettu akusto josta, riittää virtaa noin kolmeksi tunniksi, jos sähköverkosta ei saada virtaa. Tukiasemalle oli asennettu myös yksittäisiä invertterejä, joiden kautta saatiin tuotettua sähkökatkojen aikana virtaa vaihtosähkölaitteille, kuten varavoimajärjestelmän etäohjaus- ja -valvontalaitteelle.

### **5.2 Varavoimakone**

Järjestelmän varavoimakoneena käytettiin Hondan EU70is aggregaattia, joka pystyy tuottamaan jatkuvaa sähkötehoa 5,5 kW ja sen hetkittäinen maksimisähköteho on 7 kW. Aggregaatti sisälsi valmiina jo jonkin verran käyntiautomaatiikkaa. Se säättää moottorin kierrosnopeutta generaattorin kuorman mukaan, jolloin se säästää polttoainetta verrattuna aggregaattiin, joka käy vakionopeudella. Aggregaatti myös sammutuu automaattisesti, jos generaattori on ylikuormitettu, moottori ylikuumenee tai öljyä on liian vähän.

Aggregaatissa oli valmiina liitäntä etäkäynnistyslaitteelle. Liitäntä oli tarkoitettu valmistajan omalle langalliselle etäkäynnistyslaitteelle, mutta kyseistä liitäntää pystyi

hyödyntämään tämän työn etäohjauslaitteen kanssa. Liitteessä 2 on esitetty, kuinka etäohjauslaite liitettiin aggregaattiin.

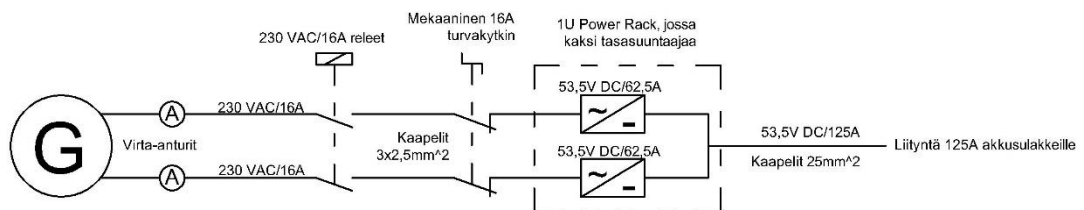
### 5.3 Virransyöttö

Varavoimakoneen 16 A:n syötöt kulkevat releiden läpi, joilla voidaan kytkeä syöttö päälle tai pois. Normaalitilanteessa releet ovat auki. Releet on sijoitettu vaihtovirtapuolelle, koska siellä virta on matalampi kuin tasavirtapuolella ja näin ollen releiden virrankesto on matalampi.

Releiden jälkeen johtimet kulkevat kuusinapaisen turvakytkimen läpi. Kyseessä on siis yksi turvakytkin, jolla voidaan katkaista kaksi syöttöä samaan aikaan. Normaalitilanteessa turvakytkin on suljettuna, mutta esimerkiksi huoltotilanteessa se aukaistaan, jolloin ei ole vahingossakaan mahdollisuutta saada generaattorilta virtaa tukiaseman järjestelmään.

Turvakytkimen jälkeen johtimet kulkevat tasasuuntaajalle. Kyseessä on yksi kehikko, johon on mahdollista kytkeä neljä kappaletta tasasuuntausmoduulia rinnan. Yhden moduulin maksimiteho on 3000 W. Tässä kyseisessä järjestelmässä käytetään kahta moduulia. Tasasuuntaajat ovat sijoitettuna tukiaseman sisälle, koska akkusulakkeetkin sijaitsevat siellä. Tasasuuntaajien on hyvä olla mahdollisimman lähellä akkusulakkeita, koska käytettävä jännite on matala ja virta korkea, jolloin on riski, että jännite putoaa johtimissa liian alhaiseksi.

Kuviossa 8 on esitetty edellä kuvatun järjestelmän tekninen piirustus.



Kuvio 8. Varavoimajärjestelmän tekninen piirustus



## 6 Tarvittavien mittausten ja ohjausten suunnittelu sekä toteutus

### 6.1 LTspice-simulointiohjelma

LTspice on Linear Technologyn valmistama ilmainen piirisimulointityökalu, jolla voidaan simuloida analogisia piirejä. Ohjelma sisältää laajan komponenttikirjaston ja kolmannen osapuolen malleja on myös runsaasti tarjolla. Simulointien avulla voidaan tutkia esimerkiksi signaalien käyttäytymistä ja mitoittaa käytettäviä komponentteja.

### 6.2 Virta-anturit

#### 6.2.1 Käytettävät virta-anturit

Järjestelmässä käytetään kahta erilaista virta-anturia. Molemmat virta-anturit ovat analogisia, joiden toimintaperiaate perustuu induktiiviseen mittaukseen.

Varavoimakoneen molempien pistokkeiden taakse asennettiin Talema AC1020-virta-anturit (ks. kuvio 9), jolla voidaan mitata maksimissaan 20 A suuruista virtaa. Tämä anturi sopii tilanteisiin, jossa mitattavan johtimen toinen pää on irrotettavissa tai johdin on mahdollista katkaista, jolloin se voidaan viedä anturissa olevan reiän lävitse. Pistokkeiden takana olevat kaapelit olivat helposti irrotettavissa, ja tämä anturi sopi kokonsa puolesta paneelin taakse. Anturin muuntosuhde on 1000:1, eli se muuntaa johtimessa kulkevan virran 1000 kertaa pienemmäksi.



Kuvio 9. Talema AC1020-virta-anturi (Partco n.d.)

Tukiaseman virransyötön mittauksessa käytetty anturi oli TDK CCT323047, jolla voidaan mitata maksimissaan 100 :n suuruista virtaa. Kyseiset anturin ovat pihtivirta-mittareita (ks. kuvio 10) eli ne saadaan avattua ja asennettua johtimen ympärille, jolloin johdinta ei tarvitse irrottaa tai katkaista. Näin ollen niiden mittauspaikkaakin voidaan muuttaa helpommin, kuin kiinteään anturiin. Anturin muuntosuhde on 3000:1, eli se muuntaa johtimessa kulkevan virran 3000 kertaa pienemmäksi.

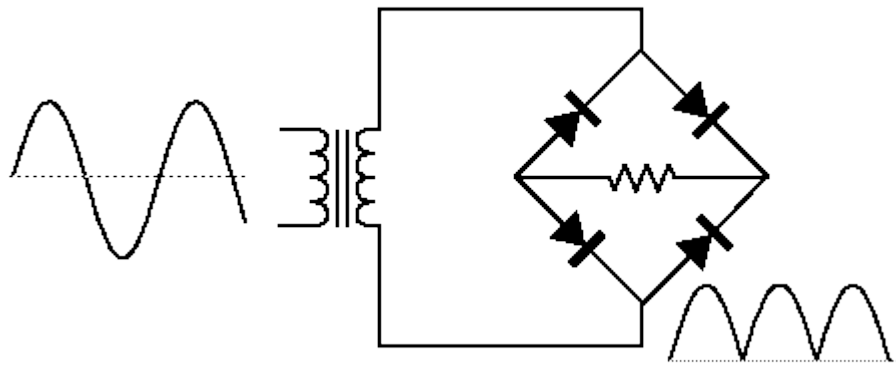


Kuvio 10. CCT323047-virta-anturi (Clamp Type AC Current Sensor 2016, 8.)

### 6.2.2 Virta-anturien signaalien tasasuuntaus

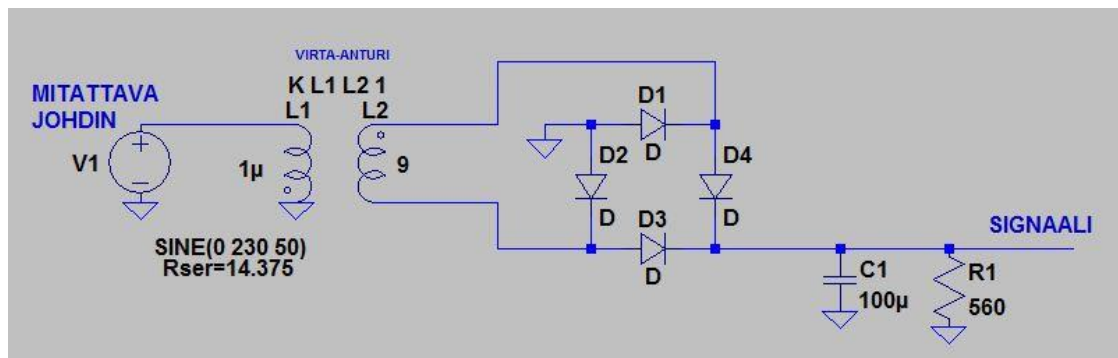
Koska kaikki mitattavat virrat ovat vaihtovirtaa, niin myös niihin indusoituva virta on vaihtovirtaa. Vaihtovirta tarvitsee tasasuunnata tasavirraksi, jotta sitä voidaan hyödyntää järkevästi.

Virta-anturin toisiopuolen johtimiin on liitetty diodisilta, joka koostuu neljästä diodista. Virta pääsee kulkemaan sillassa aina kahden diodin läpi ja kaksi muuta diodia taas estävät sen kulun. Diodisillan jälkeen signaalin negatiiviset arvot ovat suodattuneet pois, mutta virran arvo vaihtelee vielä positiivisella alueella, kuten kuviosta 11 voidaan huomata. Kyseistä suuntausta kutsutaan kokoaaltosuuntaukseksi.



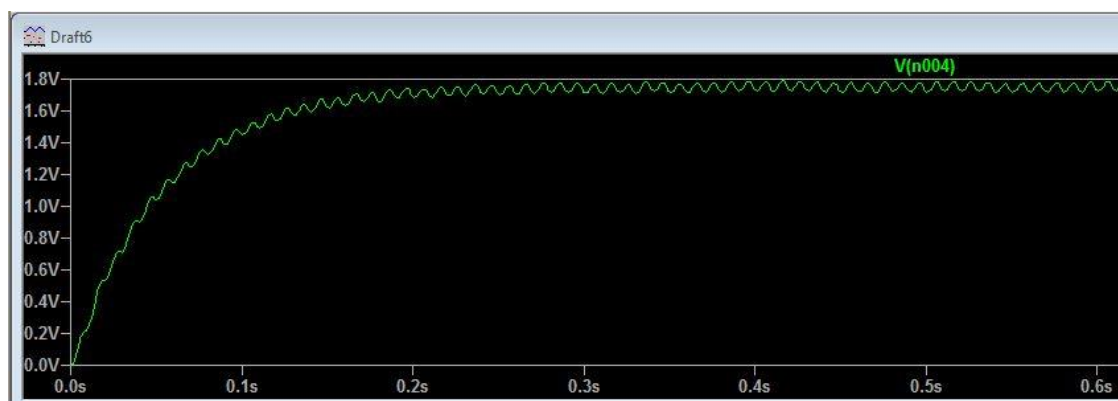
Kuvio 11. Vaihtovirran kokoaaltosuuntaus (Bridge Rectifier n.d.)

Kokoaaltosuunnatulla virralla ei vielä ole niinkään arvoa mittausteknisesti, koska virran arvo vaihtelee vielä paljon. Virtaa saadaan tasattua lisäämällä oikean suuruinen kondensaattori diodisillan jälkeen. Kondensaattori varautuu aina, kun piirin jännite on korkealla ja purkautuu, kun piirin jännite on alhaisempi kuin sen oma jännite. Kondensaattorin rinnalle kytkettiin vielä  $560\ \Omega$ :n vastus, jonka yli vaikuttavaa jännitettä mitataan. Kuviossa 12 on esitetty lopullinen mittausjärjestely tukiaseman virransyöttöä mittaavalle anturille. Generaattorin syöttöä mittaavan anturin mittausjärjestely on hyvin samanlainen (ks. liite 3).



Kuvio 12. LTspice-ohjelmalla tehty kuva tukiaseman virransyötön mittausjärjestelystä

Kuviosta 13 huomataan, että signaali ei ole täysin tasaista vaan siinä näkyy vielä hie-  
man rippelijännitettä. Rippelijännitteen suuruus on noin 0,06 V, joka on tarpeeksi  
matala tässä tapauksessa. Tarkkaa virrankulutusta näillä antureilla ei ole tarkoitus mi-  
tata vaan antureilla halutaan todentaa, tuleeko tukiasemalle virtaa sähköverkosta vai  
ei, joten pienempikin tarkkuus riittää.



Kuvio 13. Kuvakaappaus LTspice-ohjelmalla tehdystä mittausjärjestelyn simulaatiosta

### 6.3 Polttoaineen määrän mittaus

Tieto polttoaineen määrästä on hyvin oleellinen tieto erityisesti varavoimakoneille, jotta voidaan varmistua siitä, että polttoainetta on aina riittävästi tuottamaan sähkökatkon aikainen virta. Polttoaineen lisääminenkin voidaan suorittaa tehokkaammin, kun tiedetään, milloin polttoainetta alkaa olemaan liian vähän.

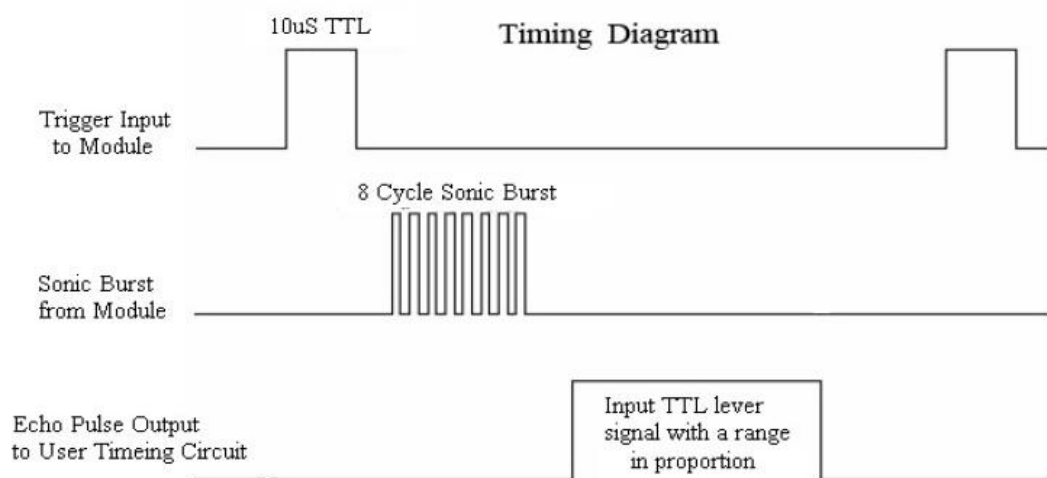
Polttoaineen määrää voidaan mitata erityyppisillä antureilla kuten paineantureilla ja ultraääniantureilla. Tässä järjestelmässä käytettävä anturi on HC-SR04-ultraäänianturi. Anturilla mitattava minimietäisyys on kaksi senttimetriä ja maksimietäisyys on 400 senttimetriä. Mittauksen tarkkuus voi parhaimmillaan olla noin kolme millimetriä, mutta tässä järjestelmässä pienempikin tarkkuus on riittävä. Anturi koostuu ultraäänilähettimestä ja -vastaanottimesta sekä ohjauspiiristä. Anturimoduulissa on liitännät käyttöjännitteelle (Vcc), pulssin lähetyksen ohjaukselle (Trig), pulssin kaiun vastaanottamiselle (Echo) sekä maalle (ks. kuvio 14).



Kuvio 14. HC-SR04 ultraäänianturi (Ultrasonic Ranging Module HC-SR04 n.d., 2.)

Raspberry Pi'n GPIO-pinnistä johdetaan virtaa 10  $\mu$ s ajan anturin Trig-pinniin (Trigger), jolloin anturin ultraäänilähetin lähettää kahdeksan syklin pulssin 40 kHz:n taajuudella (ks. kuvio 15). Samalla ohjelma tallentaa signaalin lähetyksen ajankohdan

muistiin. Kun ultraäänisignaali on törmännyt kohteeseen, eli tässä tapauksessa polttoaineen pintaan, se heijastuu takaisin anturin ultraäänivastaanottoon. Tällöin anturin ohjauspiiri lähettää tiedon Raspberry Pi'n GPIO-pinniin (kytketty anturin Echo-pinniin), joka on ohjelmoitu vastaanottamaan digitaalista tietoa. Kun tieto signaalin paluusta tulee, ohjelma tallentaa sen saapumisajan muistiin.



Kuvio 15. HC-SR04 timing diagram (Ultrasonic Ranging Module HC-SR04 n.d., 2.)

Anturin ja polttoaineen pinnan välinen etäisyys voidaan laskea kaavan 3 mukaan.

$$x = \frac{t \cdot c}{2} \quad (3)$$

missä  $x$  = anturin ja polttoaineen pinnan välinen etäisyys (m)

$c$  = 34 300 m/s (äänennopeus)

$t$  = signaalin kulkema aika (s)

Etäisyys jaetaan kahdella, koska signaali kulkee anturista kohteeseen ja takaisin, jolloin todellinen etäisyys on puolet tästä.

Polttoainetankin tilavuus ja mitat ovat tiedossa, joten polttoaineen määrä voidaan laskea anturin ja polttoaineen pinnan välisen etäisyyden avulla. Polttoainetankki on muodoltaan epäsäännöllinen monitahokas, jonka tilavuus  $V = 20$  litraa, pohjan pinta-ala  $A = 0,09\text{m}^2$  ja korkeus  $h = 0,22\text{m}$ .

Polttoaineen määrä  $V$  voidaan laskea kaavan 4 mukaisesti.

$$V = \frac{t \cdot c \cdot A}{2} \quad (4)$$

missä  $V$  = polttoaineen määrä ( $\text{m}^3$ )

$t$  = signaalin kulkema aika (s)

$c = 34\,300$  m/s (äänennopeus)

$A$  = säiliön pohjan pinta-ala

Liitteessä 6 on esitetty, kuinka polttoaineen mittausta on ohjelmoitu.

## 6.4 Käynnistysakun jännitteen mittaaminen

Generaattorin käynnistysakun jännite on yksi oleellisista tiedoista. Jos akussa ei ole tarpeeksi suurta varausta, niin varavoimakoneella ei ole tarpeeksi virtaa käynnistymiseen.

Varavoimakoneessa on 12 V:n käynnistysakku, joten sen jännite täytyy muuntaa maksimissaan 5 V:n suuruiseksi, koska AD-muuntimelle tulevat signaalit eivät voi olla tätä suurempia. Jännite saadaan laskettua tarpeeksi matalaksi kytkemällä kaksi vastusta ( $R_1 = 560\,\Omega$  ja  $R_2 = 220\,\Omega$ ) sarjaan ja ottamalla signaali näiden kahden vastuksen välistä. Varavoimakoneen testauksissa huomattiin, että akun latausjännite on noin 14 V, joten vastukset mitoitettiin varmuuden vuoksi sen mukaan, että akun latausjännite olisi 16 V. Kaavassa 5 on esitetty, kuinka suuri signaali tulee akun jännitteen ollessa 16 V.

$$V_{out} = V_{in} \frac{R_2}{R_1 + R_2} = 16\text{ V} \cdot \frac{220\,\Omega}{560\,\Omega + 220\,\Omega} = 4,5\text{ V} \quad (5)$$

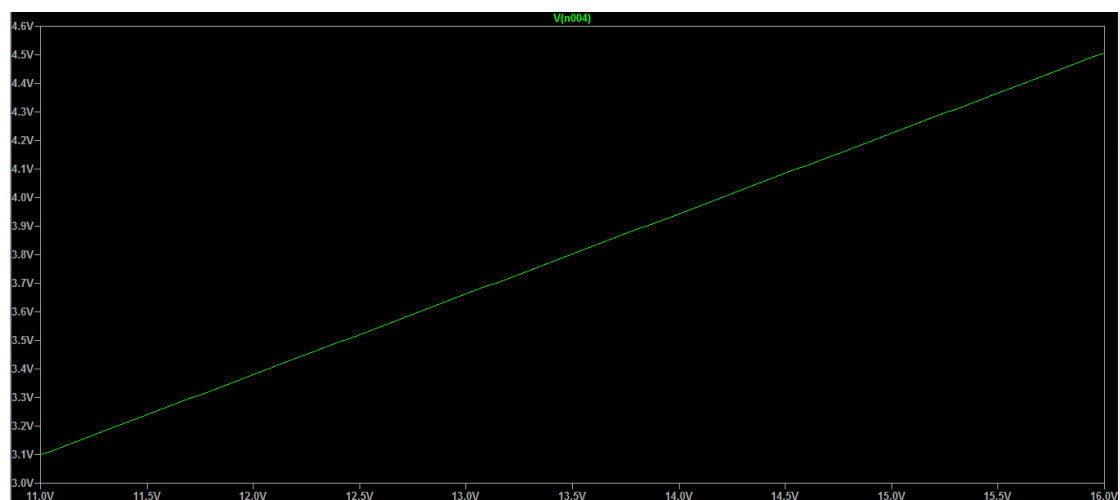
missä  $V_{in}$  = akun jännite (V)

$R_1 = 560\,\Omega$

$$R_2 = 220 \, \Omega$$

Akun jännitettä ei haluta mitata jatkuvasti, koska pienikin kuorma pitkällä aikavälillä kuluttaa akun varausta, eikä akun varaus laske niin äkillisesti, että sitä pitäisi valvoa esimerkiksi tunnin välein. Tämän takia piiriin on laitettu rele, joka on normaalitilassa auki, ja kun akun jännitettä halutaan tarkastella, rele suljetaan. Mittausjärjestely on esitetty liitteessä 4 ja liitteessä 5 on esitetty mittauksen koodi.

Kuviossa 16 on esitetty LTspice-ohjelmalla tehdyn simulaation tulos, josta voidaan lukea mitä akun jännitteen arvoa signaali vastaa.



Kuvio 16. Käynnistysakun jännitemittauksen simuloinnin tulos

## 6.5 Varavoimakoneen kotelon sisälämpötilan mittaaminen

Varavoimakoneen sisäistä lämpötilaa haluttiin myös valvoa, koska se vaikuttaa hyvin paljon varavoimakoneen suorituskykyyn. Varavoimakone itsessään sammuu automaattisesti, jos lämpötila nousee liian korkeaksi, mutta lisäämällä etäohjauslaitteis-



toon lämpötila-anturi niin saadaan tieto, että varavoimakone on sammunut juuri korkean lämpötilan takia. Anturin avulla saadaan myös tieto tukiasemalla vallitsevasta lämpötilasta, jolloin testikäynnistyksillä saadaan tietoa kuinka paljon esimerkiksi kova pakkanen vaikuttaa käynnistymiseen.

Kotelon sisälämpötilan mittaaminen suoritettiin DHT22-anturilla (ks. kuvio 17), joka on digitaalinen lämpötila- ja kosteusanturi. Kotelon sisäinen kosteus ei ollut järjestelmän kannalta niin oleellinen tieto, mutta koska anturi sisälsi valmiina molemmat ominaisuudet niin kosteuden mittaamisen mahdollisuus huomioitiin ohjelmassa. Anturi koostuu kolmesta pääkomponentista, jotka ovat lämpötilaa mittaava termistori, kapasitiivinen kosteusanturi sekä yksinkertainen A/D-piiri, joka muuntaa antureiden analogisen signaalin digitaalseksi. Kosteusmittaus ei ole varavoimajärjestelmän kannalta kovin oleellinen, mutta koska se on jo valmiina anturissa niin sitäkin käytetään. Anturilla voidaan mitata lämpötilaa  $-40 - 125\text{ }^{\circ}\text{C}$  välillä  $\pm 0,5^{\circ}\text{C}$  tarkkuudella ja suhteellista ilman kosteutta  $0 - 100\%$  välillä  $2\%$ -yksikön tarkkuudella. Anturin näytteenottotaajuus  $0,5\text{ Hz}$  on suhteellisen matala, mutta kuitenkin riittävä tämän työn tarkoitukseen. Anturin valmistaja Adafruit Industries tarjoaa sille myös avoimen lähdekoodin Python-kirjaston, jonka avulla voidaan lukea tietoa anturilta. Liitteessä 5 on esitetty lämpötilan ja kosteuden mittaamisen koodi.

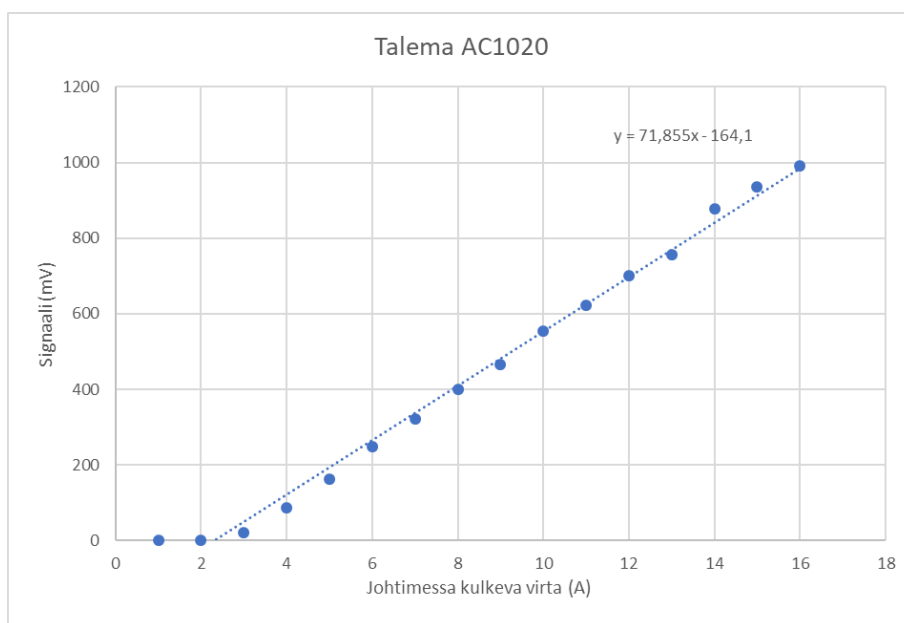


Kuvio 17. Lämpötila- ja kosteusanturi DHT22 (Digikey n.d.)

## 6.6 A/D-muunnin

Raspberry Pi ei sisällä analogisia tuloja ja suurin osa järjestelmän antureista ovat analogisia, joten tarvitaan A/D-muunnin, jotta antureiden analoginen signaali saadaan muutettua digitaaliseen muotoon. Käytettävä A/D-muunnin on 10-bittinen ja 8-kanavainen MCP3008. MCP3008 muuntaa analogisen signaalin digitaaliseen muotoon ja lähettää mittaustiedon Raspberry Pi:lle SPI (Serial peripheral interface) tiedonsiirtoväylän kautta. Koska A/D-muunnin on 10-bittinen, niin sen lähettämä tieto on lukuarvo 0 ja 1023 väliltä. Mittaustulokset halutaan kuitenkin esittää mitattavan suuren yksikössä, joten muuntimelta tuleva lukuarvo muunnetaan ohjelmassa mitattavan suureen yksikköön.

LTspice-ohjelmalla simuloitiin kuinka suuri signaali vastaa mitäkin mitattavaa virtaa. Saatuihin tuloksiin sovitettiin Excelillä trendiviiva, jonka yhtälön avulla voidaan selvittää mitä virran arvoa mikäkin signaali vastaa (ks. kuvio 18).



Kuvio 18. Generaattorin syöttöä mittaavan anturin palauttama signaali virrasta riippuen

Esimerkiksi tilanne, jossa luetaan A/D-muuntajan kanava ja sen palauttama lukuarvo on 100. Lukuarvon vastaama signaali voidaan laskea kaavalla 6.

$$V_{out} = \frac{ADC \cdot V_{ref}}{1024} = \frac{100 \cdot 5200 \text{ mV}}{1024} = 510 \text{ mV} \quad (6)$$

missä  $V_{out}$  = A/D-muuntimelle tuleva signaali (mV)

$ADC$  = A/D-muuntimelta luettava lukuarvo

$V_{ref}$  = A/D-muuntimen referenssijännite (mV)

Exceliin sovitetun trendiviivan yhtälön avulla voidaan kyseinen signaali muuttaa mitattavassa johtimessa kulkevaksi virraksi kaavan 7 mukaan.

$$I_{johdin} = \frac{V_{out} + 164,1}{71,855} = \frac{510 \text{ mV} + 164,1}{71,855} = 9,4 \text{ A} \quad (7)$$

missä  $V_{out}$  = generaattorin syöttöä mittaavan anturin signaali (mV)

Tukiaseman virransyöttöä mittaaville antureille tehtiin myös samanlainen tarkastelu ja niiden antama signaali voidaan muuttaa virraksi kaavan 8 mukaan.

$$I_{johdin} = \frac{V_{out} + 145,88}{121,29} \quad (8)$$

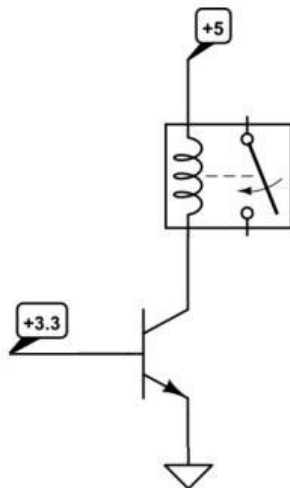
missä  $V_{out}$  = tukiaseman virransyöttöä mittaavan anturin signaali (mV)

## 6.7 Releet

Koko järjestelmän ohjaus toteutetaan neljällä releellä. Varavoimakoneen käynnistäminen suoritetaan kahdella releellä. Toinen releistä on liitettynä varavoimakoneen

päävirtapiiriin eli sillä voidaan kytkeä varavoimakoneen päävirrat päälle ja pois. Toisen releistä on taas liitettyä käynnistysmoottorin virransyöttöön, jolloin sillä voidaan ohjata starttimoottori päälle tai pois. Varavoimakone voidaan käynnistää näillä kahdella releellä tyhjäkäynnille, mutta generaattori ei vielä syötä virtaa. Generaattorin molempia syöttöjä ohjataan myös releillä. Kun varavoimakone on ollut tarpeeksi kauan tyhjäkäynnillä niin syöttöjen releet voidaan sulkea jolloin generaattori alkaa syöttämään virtaa tasasuuntaajille.

Raspberry Pi'n GPIO-pinneistä saatava jännite on vain 3,3 VDC ja käytettävien releiden kelojen ominaisjännite on 5 VDC eli releiden ohjaus ei onnistu liittämällä niitä suoraan GPIO-pinneihin. Raspberry Pi sisältää kuitenkin kaksi kappaletta 5 VDC pinniä, mutta niitä ei voi ohjata vaan ne syöttävät virtaa jatkuvasti. Tämä ongelma voidaan ratkaista käyttämällä transistoria kytkimenä. Kun transistori kantaan syötetään virtaa GPIO-pinnistä, niin se päästää lävitseen virtaa, jolloin releen kelan läpi kulkee virtaa ja releen kosketin sulkeutuu (ks. kuvio 19). Tällä tekniikalla on toteutettu järjestelmän kaikkien releiden ohjaus.



Kuvio 19. Releen ohjaus transistorin avulla

## 7 Etäohjaus- ja valvontaohjelmiston ohjelmoiminen

Etäohjaus- ja valvontaohjelmisto ohjelmoitiin Python-ohjelmointikielellä ja mittaustulokset tallennettiin CSV-tiedostoon. Yhteys etäohjauslaitteen ja toimeksiantajan tietokoneen välillä toteutettiin VNC-yhteyden avulla.

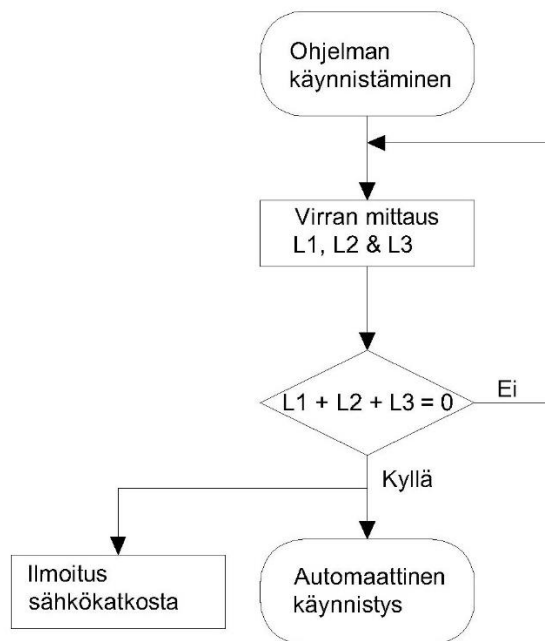
### 7.1 Automaattisen ohjelman toimintaperiaate

Ohjelma voidaan jakaa karkeasti kolmeen eri osaan:

- pääohjelma
- automaattinen käynnistys
- käyntiohjelma

#### 7.1.1 Pääohjelma

Kun sähkönsyöttö tukiasemalle toimii normaalisti niin ohjelma suorittaa niin kutsutua ”pääohjelmaa”, joka lukee puolen sekunnin välein A/D-muuntimen kanavat 0, 1 ja 2 eli ne kanavat joihin tukiaseman virransyöttöä mittaavat anturit ovat liitettynä. Ohjelma vertailee saatuja arvoja if-ehtolauseella ja jos kaikkien vaiheiden mittausten tulokset ovat nolla niin se lähettää sähköpostilla ilmoituksen sähkökatkosta ja siirtyy käynnistämään varavoimakonetta tyhjäkäynnille. Kuviossa 20 on esitetty pääohjelman vuokaavio ja liitteessä 6 on esitetty pääohjelman koodi.



Kuvio 20. Pääohjelman vuokaavio

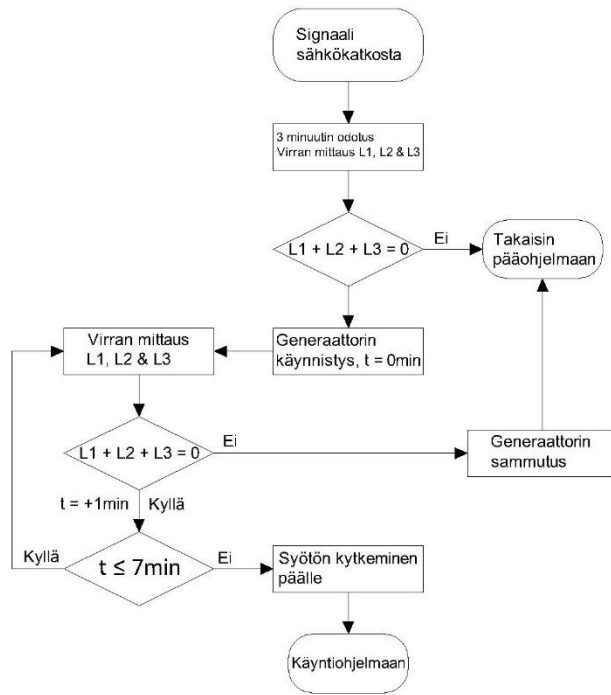
### 7.1.2 Automaattinen käynnistys

Kun ohjelma on saanut signaalin sähkökatkosta niin se siirtyy varavoimakoneen automaattiseen käynnistykseen. Ohjelma odottaa alussa kolmen minuutin verran, jonka jälkeen suorittaa jokaisen vaiheen virran mittauksen uudestaan. Kolmen minuutin odotus johtuu siitä, että sähköverkon vikatilanteissa aikajälleenkytkentä suoritetaan noin 1-3 minuutin kuluttua, jos pikajälleenkytkentä on epäonnistunut. Tällä odotuksella säästytään varavoimakoneen turhilta käynnistämisiltä. Jos odotuksen jälkeen vaiheisiin on palannut virta niin varavoimakonetta ei käynnistetä vaan ohjelma lähettää sähköpostilla ilmoituksen, että sähkökatko kesti alle kolme minuuttia ja siirtyy takaisin suorittamaan pääohjelmaa.

Mikäli kolmen minuutin jälkeen kaikki mittaukset näyttävät yhä nollaa niin ohjelma sulkee varavoimakoneen päävirtoja ohjaavan releen, jonka jälkeen odottaa kaksi sekuntia ja sulkee varavoimakoneen käynnistysmoottoria ohjaavan releen neljäksi sekunniksi.

Kun varavoimakone on käynnistetty tyhjäkäynnille niin ohjelma siirtyy seuraavaan vaiheeseen, jossa varavoimakonetta lämmitetään seitsemän minuutin verran. Vaihe toteutettiin for-silmukalla, joka suorittaa silmukan sisällä olevat komennot kunnes ennalta asetettu raja tulee täyteen (seitsemän kertaa). Jokaisella silmukan suorituskerralla se lukee A/D-muuntimen kanavat 0, 1 ja 2, ja jos kaikkien arvot ovat nolla niin se jää odottamaan minuutiksi seuraavaa mittausta ja lähettää sähköpostiin ilmoituksen kauanko varavoimakone on ollut käynnissä. Jos yhdessäkin vaiheessa havaitaan virtaa niin ohjelma aukaisee varavoimakoneen päävirtoja ohjaavan releen ja varavoimakone sammuu. Sammuttamisesta lähtee sähköpostiin ilmoitus.

Jos seitsemän minuutin jälkeen tukiaseman virransyöttö on yhä poikki niin ohjelma siirtyy kytkemään generaattorin syöttöjä päälle. Kuviosta 21 voidaan huomata, että tämä vaatii kahden ehdon täyttymisen: aikaa on pitänyt kulua seitsemän minuuttia varavoimakoneen käynnistämisestä ja virranmittausten täytyy näyttää nollaa. Ehtojen täytyttyä ohjelma sulkee ensiksi generaattorin toista syöttöä ohjaavan releen ja odottaa viisi sekuntia, jonka jälkeen sulkee myös toista syöttöä ohjaavan releen. Viiden sekunnin odotuksella saavutetaan se, että generaattorilta otetaan virtaa porrastetusti. Kuviossa 21 on esitetty automaattisen käynnistyksen vuokaavio ja liitteessä 7 on esitetty automaattisen käynnistyksen koodi.

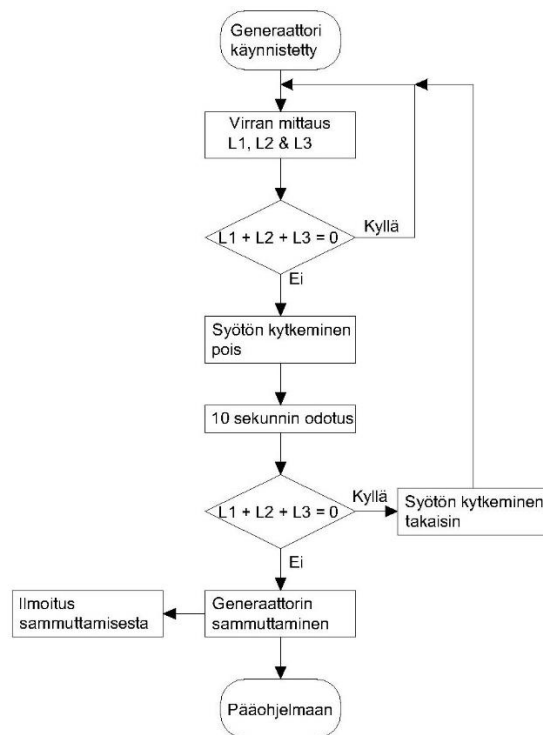


Kuvio 21. Automaattisen käynnistuksen vuokaavio

### 7.1.3 Käyntiohjelma

Käyntiohjelma mittaa myös tukiaseman kaikkien vaiheiden virtaa. Jos ohjelma havaitsee yhdessäkin vaiheessa virtaa niin se aukaisee molemmat generaattorin syöttöjä ohjaavat releet. Varavoimakonetta ei kuitenkaan sammuteta heti, vaan ohjelma odottaa 10 sekuntia jonka aikana se suorittaa virranmittauksen vielä uudestaan. Tällä lyhyellä odotusajalla vältetään varavoimakoneen turha sammuttaminen, koska sähkökatkojen aikana on yleistä, että virrat palaavat hetkittäin muutamaksi sekunniksi takaisin. Varavoimakoneen välitön sammuttaminen kuorman kanssa voi aiheuttaa myös sen, että generaattorin käämien magnetointivirta katoaa. Jos virransyöttö on edelleen pois niin generaattorin syöttö kytketään takaisin päälle ja ohjelma jatkaa normaalia suorittamistaan. Jos virransyöttö on kuitenkin palannut takaisin niin varavoimakone sammutetaan ja siirrytään takaisin pääohjelmaan. Kuviossa 22 on esitetty käyntiohjelman vuokaavio ja liitteessä 6 on esitetty sen koodi.





Kuvio 22. Käyntiohjelman vuokaavio

## 7.2 Käyntitietojen tallentaminen

Tiedot varavoimakoneen toiminnoista haluttiin tallentaa jatkokäyttöä varten. Tietojen tallennus suoritettiin hyödyntäen Pythonin pandas-kirjastoa, joka on tarkoitettu datan käsittelyyn ja sillä on mahdollista tallentaa tietoja CSV-tiedostoon. Ohjelma alkaa tallentamaan tietoja, kun varavoimakone käynnistetään tyhjäkäynnille ja lopettaa tietojen tallentamisen, sähkökatko on ohi ja varavoimakone on sammutettu.

Tallennettavia tietoja olivat:

- käynnistymis- ja sammumisaika
- käyntiaika (tyhjäkäynti ja kuormalla)
- käynnin aikainen teho
- polttoaineenkulutus
- kotelon lämpötila
- käynnistysakun latausjännite

Kuviossa 23 on esitetty Python-koodi, jolla ohjelma tallentaa mittausten tietoja.

```
def save_data():
    i = 1
    start_time = time.time()
    # Luodaan dataframe jossa ja nimetään sarakkeet
    df = pd.DataFrame(columns=['Aika (s)', 'Teho 1 (kW)', 'Teho 2 (kW)', 'Lämpötila *C', 'Akun jännite (V)',
                              'Polttoaine (l)'])
    # Ensimmäiselle riville tieto milloin varavoimakone on käynnistetty
    df.loc[0] = ["Varavoimakone käynnistetty: ", datetime.datetime.now(), "", "", "", ""]
    # Lisätään mittaustietoja sekunnin välein dataframeen niin kauan kun sähköt ovat pois
    while outage == 1:
        powers = feed_measurement()
        df.loc[i] = [round(time.time() - start_time, 2), powers[0], powers[1], temp_and_hum(value = "temp"),
                    batt_measurement(), fuel_measurement()]
        i += 1
        time.sleep(1)
    # Jos sähköt palaavat niin looppi loppuu ja varavoimakoneen sammumisaika kirjataan viimeiselle riville
    if outage == 0:
        df.loc[i] = ["Varavoimakone sammutettu: ", datetime.datetime.now(), "", "", "", ""]
        break
    # Lopuksi tallennetaan dataframe CSV-muotoon ja annetaan nimeksi kyseinen päivämäärä
    file_name = datetime.datetime.now().strftime('%Y-%m-%d') + '.csv'
    df.to_csv(file_name)
```

Kuvio 23. Funktio, jolla ohjelma tallentaa mitattavia suureita CSV-tiedostoon

### 7.3 Ohjelman käyttöliittymä ja sen hallitseminen

Raspberry Pi:hin on asennettu RealVNC-yrityksen tarjoama VNC-palvelinohjelma, johon voidaan muodostaa etätyöpöytäyhteys toiselta laitteelta käyttäen saman yrityksen VNC-pääteohjelma. Etäohjaus- ja valvontaohjelma on käynnissä Raspberry Pi:n työpöydällä ja sen käyttöliittymää voidaan käyttää muodostamalla VNC-yhteys toiselta laitteelta siihen. Raspberry Pi:hin liitettiin normaali USB-modeemi, jolla se muodosti yhteyden internettiin.

Graafisesta käyttöliittymästä ei ollut tarkoitus tehdä visuaalisesti näytävää, koska järjestelmän on tarkoitus toimia autonomisesti ilman käyttäjää, jolloin graafisen käyttöliittymän käyttö jää hyvin pieneksi. Käyttöliittymä toteutettiin Pythonin Tkinter-paketilla, jolla on mahdollista tehdä graafisia käyttöliittymiä.

Graafinen käyttöliittymä sisältää painikkeet, joilla voidaan manuaalisesti ohjata varavoimakoneen päävirtaa, käynnistysmoottoria, generaattorin syöttöjä sekä käynnistysakun jännitteen mittausta. Manuaalisen ohjauksen lisäksi painikkeiden tiloista voidaan nähdä, onko kyseinen rele auki vai kiinni. Käyttöliittymästä voidaan lukea myös kaikkien mittausten tulokset, jotka päivittyvät puolen sekunnin välein. Kuviossa 24 on

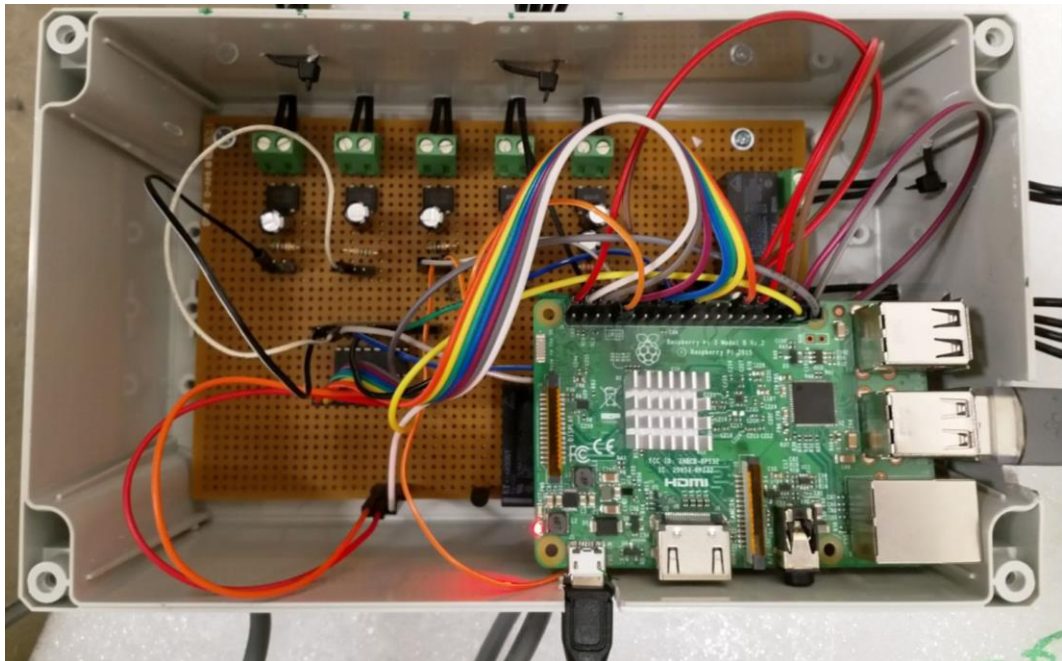
esitetty ohjelman lopullinen käyttöliittymä ja liitteessä 8 kuinka mittausten esittäminen on ohjelmoitu.



Kuvio 24. Etäohjaus- ja valvontaohjelman käyttöliittymä tilanteessa, jossa varavoimakone ei ole käynnissä

## 8 Tulokset

Opinnäytetyön lopputuloksena syntyi varavoimajärjestelmän etäohjaus- ja -valvontalaitteiston prototyyppi (ks. kuvio 25). Ominaisuuksiltaan laitteisto toteutti kaikki sille työn alussa määritellyt tavoitteet. Laitteistoa koekäytettiin varavoimajärjestelmän kanssa ensiksi testikäytössä toimeksiantajan tiloissa simuloimalla sähkökatkoja ja tekemällä manuaalisia käynnistyskäyntejä. Testaukset osoittivat koko järjestelmän toimivan hyvin luotettavasti erilaisissa tilanteissa, joten se voitiin toimittaa pilotointikohteeseen, jossa sitä päästiin testaamaan oikeassa ympäristössään.



Kuvio 25. Työn tuloksena syntynyt prototyyppi

Alkuun laitteistoa testattiin siten, että siinä pyöri vain mittaukset eikä ohjelmalla ollut mahdollisuutta ohjata varavoimajärjestelmää. Tällä vaiheittaisella testaamisella haluttiin varmistua, että internetyhteys on luotettava, jotta Raspberry Pi:hin saatiin aina tarvittaessa muodostettua yhteys. Toinen epävarma tekijä oli Raspberry Pi:n lämpeneminen tukiaseman laitesuojassa, koska siellä voi ajoittain olla hyvinkin lämmin. Kun varmistuttiin, että internetyhteys tai lämpötila ei ole ongelma niin järjestelmää päästiin testaamaan siten, että kaikki sen ominaisuudet olivat käytössä. Tukiasemalla järjestelmää testattiin siten, että asemalle tuleva virransyöttö katkaistiin manuaalisesti, jolloin tilanne muistutti oikeata sähkökatkoa. Varavoimakone käynnistyi suunnitellusti ja muutkin ohjaukset toimivat niin kuin pitikin. Liitteessä 9 on esitetty testauksen yhteydessä sähköpostiin tulleet ilmoitukset, liitteessä 10 on esitetty kyseisen testauksen aikana tallennetut mittaustiedot ja liitteessä 11 on esitetty lopulliset Raspberry Pi:n kytkennät.

## 9 Pohdinta ja jatkotoimenpiteet

Opinnäytetyön tavoitteena oli suunnitella ja toteuttaa varavoimajärjestelmän etäohjaus- ja -valvontalaitteiston prototyyppi, jonka tuli itsenäisesti ohjata varavoimakonetta mahdollisien sähkökatkojen aikana. Laitteiston kautta tuli olla myös mahdollisuus suorittaa manuaalisia testiajoja varavoimajärjestelmälle. Ohjelmiston tuli sisältää jonkinlainen käyttöliittymä, jonka kautta käyttäjän on mahdollista tarkistaa varavoimajärjestelmään asennettujen mittausten arvoja.

Työn toteutus sujui yllättävänkin mutkattomasti ja kaikki saatiin suoritettua aikataulussa. Toimeksiantajayrityskin oli tyytyväinen työn lopputulokseen. Kehitetty prototyyppi sisälsi kaikki ominaisuudet, jotka toimeksiantajayritys oli sille työn alussa määrittänyt. Määritetyt ominaisuudet olivat

- varavoimakoneen automaattinen käynnistäminen sähkökatkojen ajaksi
- ilmoitukset käyttäjälle varavoimakoneen toiminnoista sähkökatkojen ajalta
- mahdollisuus suorittaa manuaalisia testiajoja etänä
- eri mittauksia
  - sähköverkosta tukiasemalle syötettävä virta
  - generaattorin syöttämä teho
  - jäljellä oleva polttoaine
  - varavoimakoneen kotelon sisäinen lämpötila
  - varavoimakoneen käynnistysakun jännite
- käyttöliittymä, josta on mahdollista lukea mitattavat suureet sekä suorittaa manuaaliset testaukset

Vaikka laitteisto toteuttikin sille alussa määritellyt tavoitteet niin työn varrella heräsi useita arvokkaita jatkokehitysideoita, jos vastaavaa laitteistoa halutaan käyttää jatkossa laajemmassa mittakaavassa. Yksi suurin jatkotutkimuksen kohde olisi etsiä Raspberry Pi:lle toinen vaihtoehto, koska kyseinen laite on tarkoitettu enemmän prototyyppien tekemiseen, kuin tuotteessa toimimiseen. Joten sille olisi hyvä löytää laadukkaampi vaihtoehto. Etäohjaus- ja -valvontalaitteen ja käyttäjän laitteen välinen yhteys olisi parempi toteuttaa myös eri tekniikalla, koska sujuva VNC-yhteyden käyttäminen vaatii suhteellisen nopeaa internet-yhteyttä, joka taas nostaa laitteiston kustannuksia. Prototyypissä VNC-yhteys on hyvä ratkaisu, koska sitä kautta pääsee muokkaamaan mm. laitteen asetuksia, mutta lopullisessa laitteistossa tälle ei ole tar-

vetta. Osa työssä käytetyistä antureista olivat myös tarkoitettu enemmän prototyyppien valmistamiseen, joten niillekin voisi etsiä laadukkaampia vaihtoehtoja. Vaikkakin komponentteja korvattaisiin laadukkaimmilla niin sen vuoksi ei tarvitsisi tehdä paljoa lisätyötä, koska niidenkin toimintaperiaatteet ovat samanlaiset kuin työssä käytettyjen. Muutenkin etäohjaus- ja -valvontalaitteiston kotelointiin ja kaapelointiin voisi panostaa vielä enemmän, jos tulevaisuudessa kyseisiä laitteita asennettaisiin enemmänkin, koska se helpottaisi huomattavasti laitteiston asennusta sekä huoltoa.

## Lähteet

Bolton, W. 2009. Programmable Logic Controllers (5<sup>th</sup> Edition). Oxford: Elsevier.

Bridge Rectifier. N.d. Opiskelumateriaali tasasuuntaustekniikoista. Georgia State University, Department of Physics and Astronomy. Viitattu 23.4.2018.

<http://hyperphysics.phy-astr.gsu.edu/hbase/Electronic/rectbr.html>

Cheatle, K.R. 2006. Fundamentals of Test Measurement Instrumentation. Durham: ISA-Instrumentation, Systems and Automation Society.

Clamp Type AC Current Sensors. 2016. For energy management systems. Virta-anturien datalehti. TDK.

Cornell, C. 2012. Control Systems Engineer Technical Reference Handbook. Durham: International Society of Automation.

Digikey. N.d. Elektroniikkakomponentteja myyvän yrityksen verkkokauppa. Viitattu 9.5.2018. <https://www.digikey.com/catalog/en/partgroup/dht22-temperature-and-humidity-sensor/58084>

Generator EU70is Owner's Manual. N.d. Honda aggregaatin käyttöohje. Honda

GPIO. N.d. GPIO-pinnien ominaisuuksien esittely Raspberry Pi'n sivustolla. Viitattu 17.4.2018. <https://www.raspberrypi.org/documentation/usage/gpio/README.md>

Gupta, A.K., Arora, S.K., Westcott, J.R. 2017. Industrial Automation and Robotics. Dulles: Mercury Learning and Information.

Hughes, T.A. 2015. Measurement and Control Basics (5<sup>th</sup> edition). Durham: International Society of Automation.

Iwansson, K., Sinapius, G. & Hoornaert, W. 1999. Handbook Of Sensors and Actuators 7: Measuring Current, Voltage and Power. Amsterdam: Elsevier.

Kainuussa mahdollisia häiriöitä hätänumeroon soittaessa – voi koettaa soittaa ilman sim-korttia. 30.12.2017. Ylen artikkeli Kainuun alueen televerkon häiriöistä. Viitattu 20.5.2018. <https://yle.fi/uutiset/3-10000701>

Keskeytystilasto 2016. 2017. Sähköverkkojen keskeytystilasto vuodelta 2016. Helsinki: Energiateollisuus ry ET. Viitattu 29.4.2018. [https://energia.fi/files/1670/Sahkon\\_keskeytystilasto\\_2016.pdf](https://energia.fi/files/1670/Sahkon_keskeytystilasto_2016.pdf)

Mitä tarkoittavat PJK ja AJK?. 21.11.2017. Artikkelijälleenkytkennöistä Fingridin sivustolla. Viitattu 29.4.2018. <https://www.fingridlehti.fi/mita-tarkoittavat-pjk-ja-ajk/>

Opi Pythonia. N.d. Python verkko-opiskelumateriaalia. Viitattu 20.5.2018. <https://linkki.github.io/python2017/perusasiat.html>

Partco. N.d. Elektroniikkakomponentteja myyvän yrityksen verkkokauppa. Viitattu 23.4.2018. <https://www.partco.fi/fi/elektronikan-komponentit/anturit/saehkoe/14000-mmac-20a.html>

Ojala, S. 2018. Pitkät sähkökatkot paljastivat kännykkäverkon haavoittuvuuden – kävi tuuri, ettei sattunut tulipaloja tai sairauskohtauksia. Helsingin Sanomat 31.1.2018. Viitattu 20.5.2018. <https://www.hs.fi/kotimaa/art-2000005546045.html>

Principle of current sensor. N.d. Artikkelin Hiokin verkkokaupan sivustolla. Viitattu 18.4.2018. <https://www.hioki.com/en/products/listUse/?category=17>

Rasila, A. 2014. Helsingin yliopiston oppimateriaali Python-ohjelmointikieleen. Helsinki: Solmu.

Raspberry Pi Pinout Diagram | Circuit Notes. N.d. Raspberry Pi'n esittely elktoriikkaverkkokaupan sivuilla. Viitattu 21.5.2018. <https://www.jameco.com/Jameco/workshop/circuitnotes/raspberry-pi-circuit-note.html>

Raspberry Pi 3 Model B. N.d. Raspberry Pi Model 3 B'n tuote-esittely valmistajan verkkosivulla. Viitattu 18.4.2018. <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>

Richardson, T., Stafford-Fraser, Q., Wood, K. R. & Hopper, A. 1998. Virtual Network Computing. IEEE. <http://www.cl.cam.ac.uk/research/dtg/www/publications/public/files/tr.98.1.pdf>

ST 52.30.01. 2016. Akkuhuoneet ja varaamotilat. Espoo: Sähkötieto ry.

Säe, J. 2017. Aspects of Critical Communications in Disturbance Scenarios. Väitöskirja. Tampere: Tampereen teknillinen yliopisto. Viitattu 26.4.2018. <http://urn.fi/URN:ISBN:978-952-15-3933-6>

Thompson, L. M. 2006. Basic Electricity And Electronics For Control: Fundamentals And Applications (3<sup>rd</sup> Ed). Durham: ISA-Instrumentation, Systems and Automation Society.

Transistori. N.d. Sähkötekniikan oppimateriaalia peda.net sivustolla. Viitattu 20.5.2018. <https://peda.net/oppimateriaalit/e-oppi/verkkokauppa/yl%C3%A4koulu/poistuneet-tuotteet/e9k1/3srek/transistori>

Ultrasonic Ranging Module HC-SR04. N.d. Ultraäänianturin datalehti. Elec Freaks.

Varavoimalaitokset. 2013. ST-käsikirja 31. Espoo: Sähkötieto ry.

Varmennetut sähkönjakelujärjestelmät. 2005. ST-käsikirja 20. Espoo: Sähkötieto ry.

VNC User Guide 5.3. 2015. RealVNC-ohjelman käyttöohje. Cambridge: RealVNC.

VNC (Virtual Network Computing). N.d. Ohjeet VNC-ohjelman käyttöönottoon Raspberry Pi:lla. Viitattu 20.5.2018. <https://www.raspberrypi.org/documentation/remote-access/vnc/README.md>

Warne, D.F. 2005. Newnes Electrical Power Engineer's Handbook (2<sup>nd</sup> Edition). Oxford: Elsevier.

Welcome to Raspbian. N.d. Raspbian käyttöjärjestelmän esittely Raspbianin verkkosivulla. Viitattu 23.4.2018. <https://www.raspbian.org/FrontPage>



What is Python? N.d. Python ohjelmointikielen esittely Pythonin verkkosivulla. Viitattu 20.5.2018. <https://www.python.org/doc/essays/blurb/>

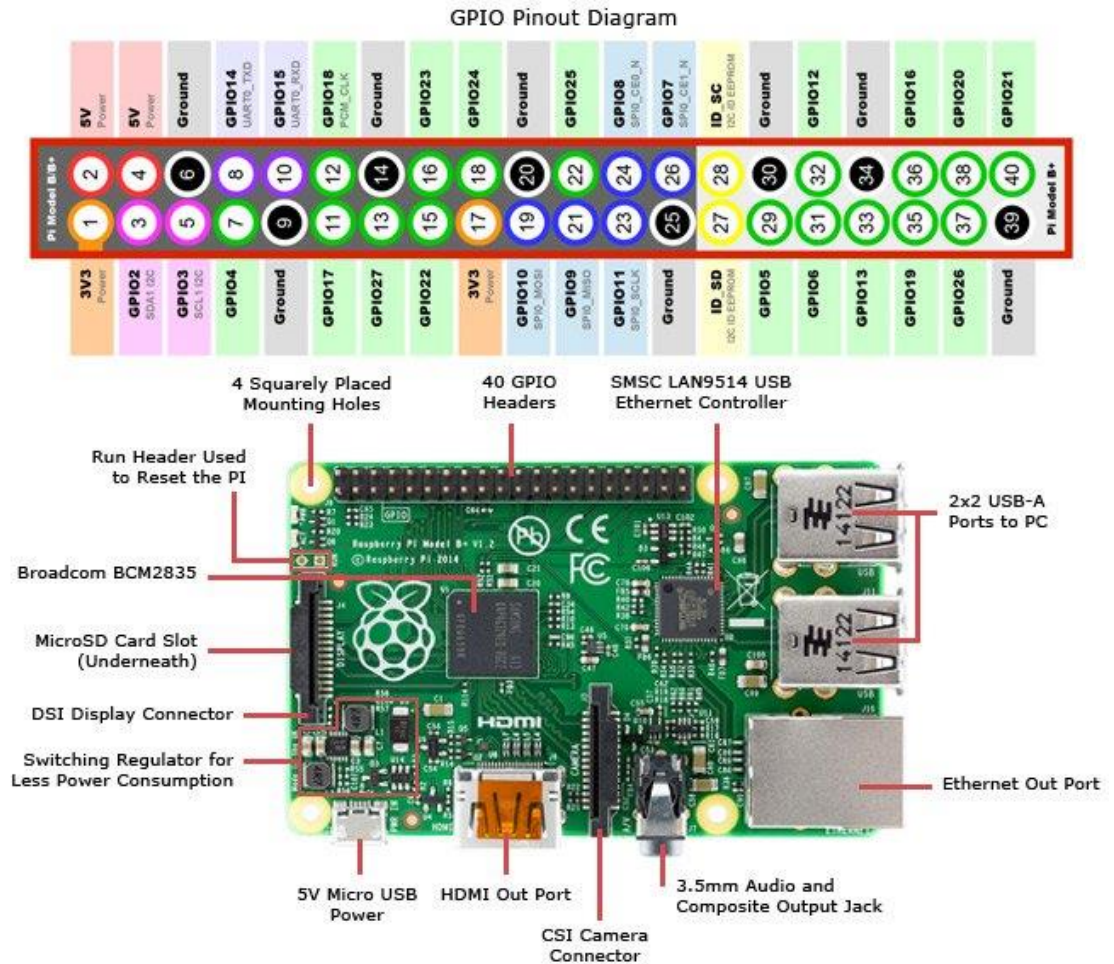
What is a Raspberry Pi? N.d.a. Artikkelin OpenSource.com sivustolla. Viitattu 9.2.2018. <https://opensource.com/resources/raspberry-pi>

What is a Raspberry Pi? N.d.b. Artikkelin Raspberry Pin sivustolla. Viitattu 9.2.2018. <https://www.raspberrypi.org/help/what-%20is-a-raspberry-pi/>

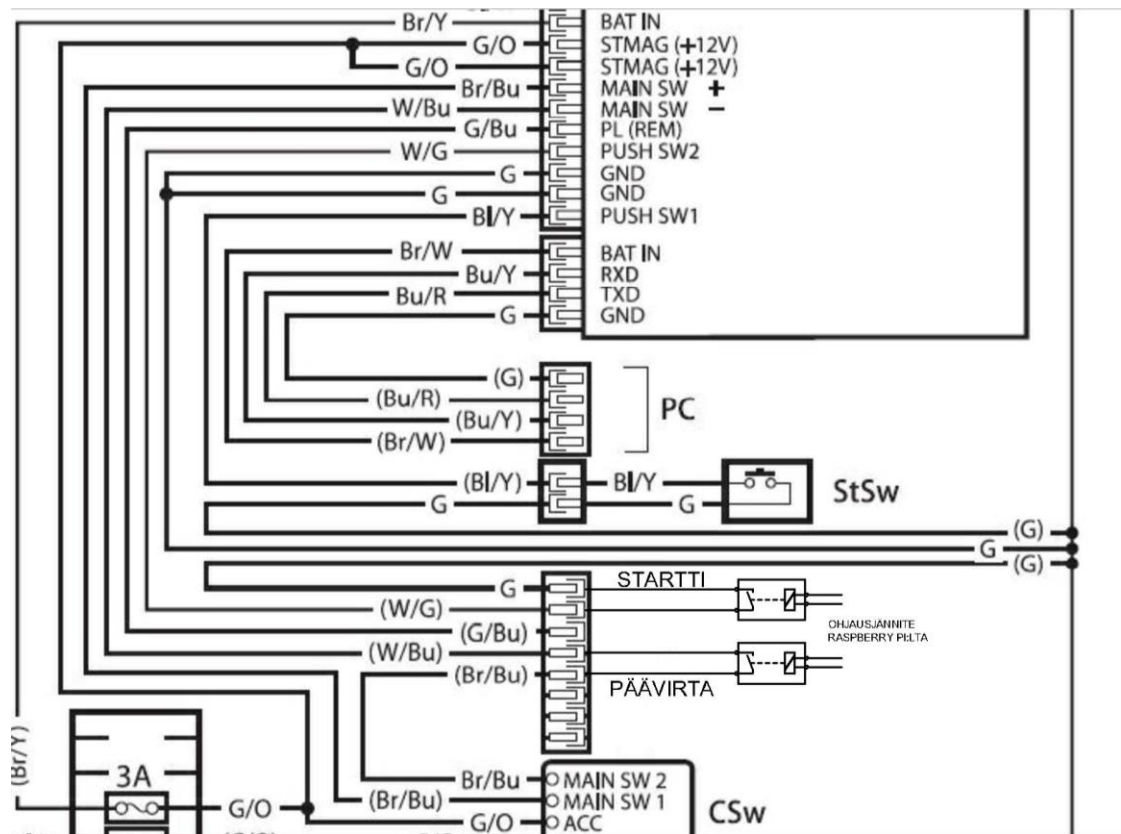
Yleistietoa häiriöistä. N.d. Energiatieto ry:n tietopaketti sähkökatkoista. Viitattu 29.4.2018. [https://energia.fi/perustietoa\\_energia-alasta/energiaverkot/sahkokatkot](https://energia.fi/perustietoa_energia-alasta/energiaverkot/sahkokatkot)

## Liitteet

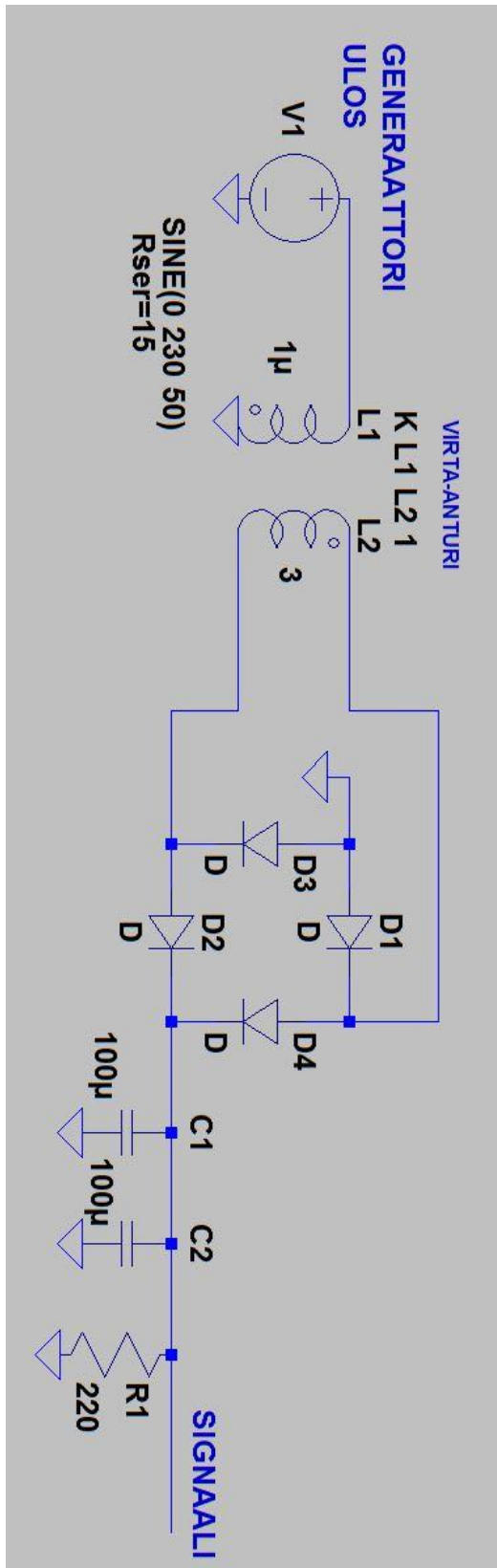
Liite 1. Raspberry Pi Model 3 B'n liittännät (Raspberry Pi Pinout Diagram | Circuit Notes. N.d.)



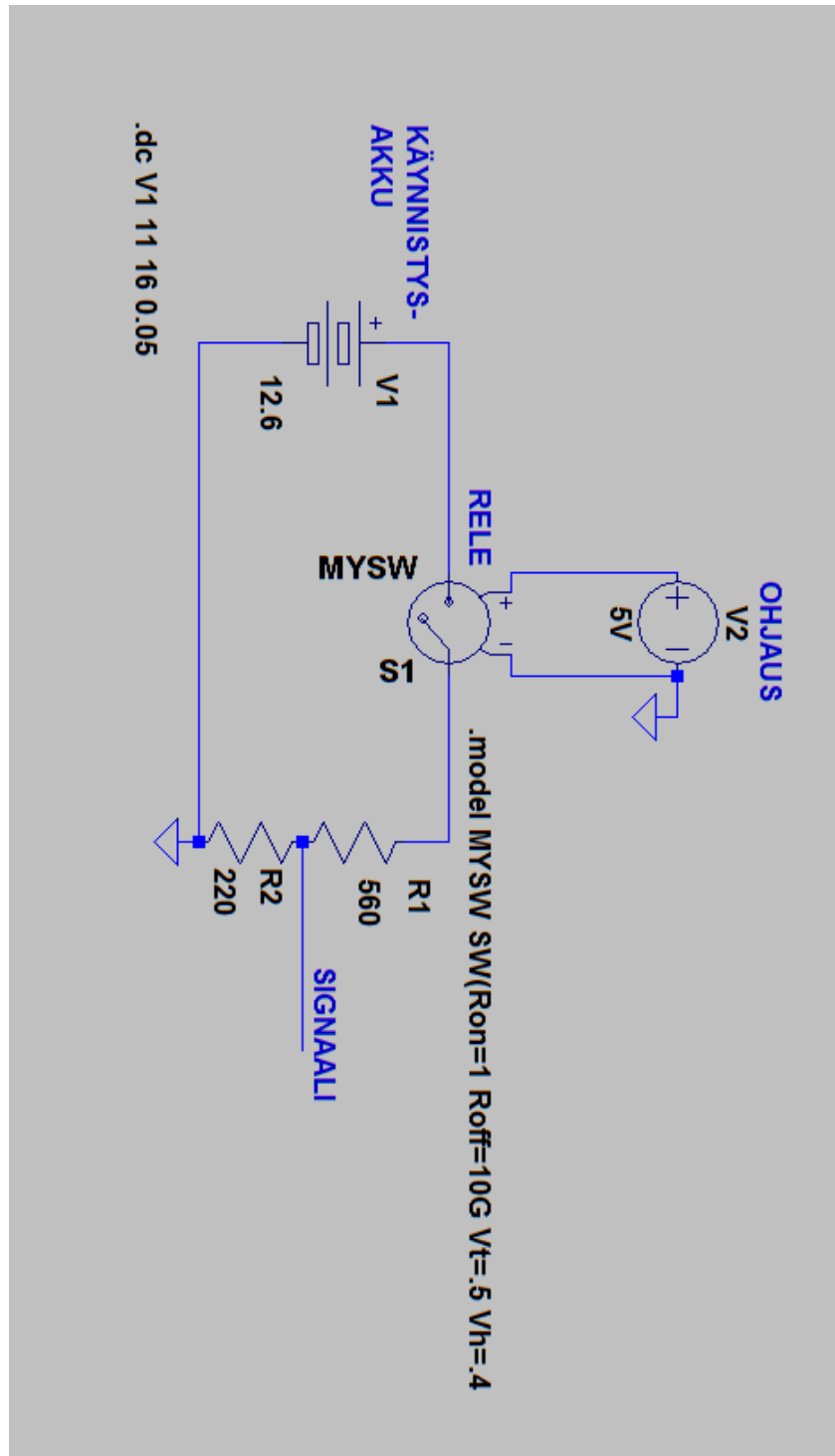
Liite 2. Etäohjauksen kytkentä (Generator EU70is Owner's Manual. N.d. muokattu)



Liite 3. Generaattorin syötön virranmittauksen mittausjärjestely



Liite 4. Käynnistysakun jännitemittauksen mittausjärjestely



Liite 5. Käynnistysakun, lämpötilan, kosteuden ja polttoaineen mittausten koodi

```
def batt_measurement():  
    # Tekee viisi mittausta ja palauttaa niiden keskiarvon, koska jännite mittauksen arvo heiluu välillä  
    voltages = []  
    for i in range(1, 5):  
        voltages.append((mcp.read_adc(7) * 5.2 * (220 + 560)) / 1024 / 560)  
    avg_voltage = round(sum(voltage) / float(len(voltage)), 2)  
    return avg_voltage  
  
def temp_and_hum(value):  
    # Sensorin lukeminen  
    humidity, temperature = Adafruit_DHT.read_retry(sensor, pin)  
    # Palauttaa arvon riippuen annetusta argumentista  
    if value == "both":  
        return humidity, temperature  
    elif value == "temp":  
        return temperature  
    elif value == "hum":  
        return humidity  
  
def fuel_measurement():  
    # Pulsin lähetys  
    GPIO.output(15, HIGH)  
    time.sleep(0.00001)  
    GPIO.output(15, LOW)  
    while GPIO.input(14) == 0:  
        start_time = time.time()  
    while GPIO.input(14) == 1:  
        stop_time = time.time()  
    # Pulsin kulkema aika  
    time_elapsed = stop_time - start_time  
    # Anturin ja pinnan välinen etäisyys  
    distance = (time_elapsed * 34300) / 2  
    # Polttoaineen määrän laskeminen  
    fuel_amount = round(0.09 * distance * 1000, 1)  
    # Funktiota kutsuttaessa se palauttaa polttoaineen määrän litroissa  
    return fuel_amount
```

## Liite 6. Pää- ja käyntiohjelman koodi

```

class Logic(threading.Thread):
    def run(self):
        # 0 = ei sähkökatkoa
        # 1 = sähkökatko todettu
        outage = 0
        while True:
            # Kutsutaan funktiota, joka palauttaa listan, jossa on jokaisen vaiheen mittauksen arvo ja
            # nimeään ne
            phases = grid_measurement()
            I1 = phases[0]
            I2 = phases[1]
            I3 = phases[2]

            # Jos missikään vaiheesta ei kulje virtaa eikä sähkökatkoa ole aikaisemmin todettu niin
            # serveritilanne käynnistetään
            if I1 == 0 and I2 == 0 and I3 == 0 and outage == 0:
                gen_auto_on()
                outage = 1

            # Jos yhdestäkään vaiheesta kulkee vähintään virtaa ja sähkökatko on aikaisemmin todettu niin
            # serveritilanne sammutetaan
            elif I1 != 0 or I2 != 0 or I3 != 0:
                if outage == 1:
                    gen_off()
                    emailer("Virrat palanneet tukiasemalla. Sammutetaan generaattori")
                    outage = 0

            # Tilannetta seurataan puolen sekunnin välein
            time.sleep(0.5)

```

## Liite 7. Varavoimakoneen automaattisen käynnistyksen koodi

```

def gen_auto_on():
    # Sähköpostiin lähetetään ilmoitus sähkökatkosta ja ohjelma jää odottamaan kolmeksi minuutiksi
    emailer("Tukiasemalla havaittu sähkökatko")
    time.sleep(180)
    # Jos kolmen minuutin odotuksen jälkeen kaikki A/D-muuntimen kanavat näyttävät nollaa niin sähköpostiin lähetetään ilmoitus,
    # että katko on kestänyt yli kolme minuuttia ja varavoimakone käynnistetään. Jos taas kolmen minuutin jälkeen
    # jossakin vaiheessa on virtaa niin sähköpostiin lähetetään ilmoitus, että varavoimakonetta ei käynnistetty
    if mcp.read_adc(0) == 0 and mcp.read_adc(1) == 0 and mcp.read_adc(2) == 0:
        emailer("Katko kestänyt kolme minuuttia. Käynnistetään generaattori tyhjäkäynnille")
        gen_start()
    else:
        emailer("Sähkökatko kesti alle kolme minuuttia, generaattoria ei käynnistetty")

def gen_start():
    # Suljetaan varavoimakoneen päävirtoja ohjaava rele
    main.on()
    # Ohjelma odottaa kaksi sekuntia ennen kuin sulkee starttimootoria ohjaavan releen
    time.sleep(2)
    start.on()
    # Starttimootoria ohjaava rele pidetään suljettuna neljä sekuntia jonka jälkeen se aukaistaan
    time.sleep(4)
    start.off()
    # Jos virrat ovat yhä pois niin siirrytään lämmittämään varavoimakone
    if mcp.read_adc(0) == 0 and mcp.read_adc(1) == 0 and mcp.read_adc(2) == 0:
        gen_warm_up()
    # Jos taas jossain vaiheessa havaitaan virtaa niin varavoimakone sammutetaan
    else:
        gen_off()

def gen_warm_up():
    # Varavoimakonetta pidetään tyhjäkäynnillä enintään 7 minuuttia
    for i in range(1, 7):
        # Verkon tilanne tarkistetaan minuutin välein
        if mcp.read_adc(0) == 0 and mcp.read_adc(1) == 0 and mcp.read_adc(2) == 0:
            time.sleep(60)
            emailer("Varavoimakone ollut käynnissä " + str(i) + " minuuttia.")
        # Jos virrat palaavat niin varavoimakone sammutetaan
        else:
            emailer("Virrat palanneet tukiasemalle. Varavoimakone sammutetaan.")
            gen_off()
            break
    # Jos 7 minuutin jälkeen virrat ovat yhä pois niin ohjelma kytkee generaattorin syötöt päälle
    if mcp.read_adc(0) == 0 and mcp.read_adc(1) == 0 and mcp.read_adc(2) == 0:
        feed_1.on()
        # Syöttöjen kytkemisen välissä pieni viive, jotta generaattorilta ei oteta suoraan kaikkea virtaa
        time.sleep(5)
        feed_2.on()

```



## Liite 8. Käyttöliittymän mittausten koodi

```
class Gui(threading.Thread):
    def run(self):
        while True:
            # Kutsutaan funktiota, joka palauttaa listan, jossa on jokaisen vaiheen mittauksen arvo ja
            # päivitetään ne käyttöliittymään
            phases = grid_measurement()
            L1_var.set(str(round(phases[0], 2)) + " A")
            L2_var.set(str(round(phases[1], 2)) + " A")
            L3_var.set(str(round(phases[2], 2)) + " A")

            # Kutsutaan funktiota, joka palauttaa listan, jossa on kunkin syöttöjohdon mittauksen arvot ja
            # päivitetään ne käyttöliittymään
            feeds = feed_measurement()
            feed_var.set("OUTPUT_1: " + str(feeds[0]) + " kW\n" + "OUTPUT_2: " + str(feeds[1]) + " kW")

            # Käynnistysajan lähittimen päivittäminen käyttöliittymään
            battery_voltage.set(str(batt_measurement()) + " V")

            # Rasperry Pin prosessorin lämpötilan päivittäminen käyttöliittymään
            cpu_temp.set("PROSESSORIN LÄMPÖTILA: \n" + os.popen("vcgencmd measure_temp").readline())

            # Koteloon sisäisen lämpötilan ja kosteuden päivittäminen käyttöliittymään
            humidity, temperature = Adafruit_DHT.read_retry(sensor, pin)
            temp_and_hum.set("Lämpötila = {0:0.1f}°C\n" + "\n" + "Kosteus = {1:0.1f}%".format(temperature,
                                                                                          humidity))

            # Polttoaineen määrä päivittäminen käyttöliittymään
            fuel_var.set("Polttoainetta = " + str(fuel_measurement()) + " l")

            # Viennut päivittyvät 1 sekunnin välein
            time.sleep(0.5)
```

Liite 9. Testiajon aikana tulleet sähköposti-imoitukset

Ensimmäinen		Sosiaaliset verkostot	Tarjoukset	+	
<input type="checkbox"/>	☆ minä	TUKIASEMA 1 - Käyntiaika: 19 min, josta kuormalla 12 min. Polttoainetta jäljellä: 18,9 L. Akku: OK			14.58
<input type="checkbox"/>	☆ minä	TUKIASEMA 1 - Sähkökatko loppunut. Kytketään syöttö pois ja sammutetaan varavoiimakone			14.58
<input type="checkbox"/>	☆ minä	TUKIASEMA 1 - Generaattorin syöttö 2 kytketty paalle			14.46
<input type="checkbox"/>	☆ minä	TUKIASEMA 1 - Generaattorin syöttö 1 kytketty paalle			14.46
<input type="checkbox"/>	☆ minä	TUKIASEMA 1 - Varavoiimakone ollut käynnissä 7 minuuttia.			14.46
<input type="checkbox"/>	☆ minä	TUKIASEMA 1 - Varavoiimakone ollut käynnissä 6 minuuttia.			14.45
<input type="checkbox"/>	☆ minä	TUKIASEMA 1 - Varavoiimakone ollut käynnissä 5 minuuttia.			14.43
<input type="checkbox"/>	☆ minä	TUKIASEMA 1 - Varavoiimakone ollut käynnissä 4 minuuttia.			14.42
<input type="checkbox"/>	☆ minä	TUKIASEMA 1 - Varavoiimakone ollut käynnissä 3 minuuttia.			14.41
<input type="checkbox"/>	☆ minä	TUKIASEMA 1 - Varavoiimakone ollut käynnissä 2 minuuttia.			14.40
<input type="checkbox"/>	☆ minä	TUKIASEMA 1 - Varavoiimakone ollut käynnissä 1 minuuttia.			14.39
<input type="checkbox"/>	☆ minä	TUKIASEMA 1 - Varavoiimakone käynnistetty			14.38
<input type="checkbox"/>	☆ minä	TUKIASEMA 1 - Katko kestänyt yli 3 minuuttia. Käynnistetään varavoiimakone tyhjäkäynnille			14.38
<input type="checkbox"/>	☆ minä	TUKIASEMA 1 - Tukiasemalla havaittu sähkökatko			14.35

## Liite 10. Testiajon aikana tallennetut mittaustiedot

	Aika (s)	Teho 1 (kW)	Teho 2 (kW)	Lämpötila (*C)	Akun jännite (V)	Polttoaine (l)
	Varavoimakone käynnistetty:			2018-04-25 09:12:26.628388		
Tyhjäkäyntiä	0	0	0	11,2	12,55	20,1
	1	0	0	11,2	12,55	20,1
	2	0	0	11,2	12,55	20,1
	3	0	0	11,2	12,55	20,1
	5	0	0	11,2	12,55	20,1
	6	0	0	11,2	12,56	20,1
	7	0	0	11,2	12,56	20,1
	8	0	0	11,2	12,55	20,1
	9	0	0	11,2	12,57	20,1
	10	0	0	11,2	12,61	20,1
	11	0	0	11,2	12,61	20,1
	12	0	0	11,2	12,62	20,1
	13	0	0	11,2	12,62	20,1
	14	0	0	11,3	12,62	20,1
	15	0	0	11,2	12,61	20,1
	16	0	0	11,3	12,62	20,1
	Välistä poistettu dataa, jotta mittauksen alku ja loppu mahtuu samalle sivulle					
Kuormalla	1130	2,21	2,15	35,8	13,82	19
	1131	2,13	2,23	35,8	13,82	18,9
	1132	2,18	2,23	35,6	13,83	18,9
	1133	2,18	2,11	35,8	13,82	18,8
	1134	2,21	2,25	35,8	13,82	18,9
	1135	2,12	2,32	35,8	13,83	18,8
	1136	2,17	2,11	35,8	13,83	18,9
	1137	2,26	2,13	35,5	13,83	18,9
	1138	2,26	2,27	35,8	13,83	18,9
	1139	2,13	2,11	35,6	13,83	19
	1140	2,11	2,35	35,8	13,83	18,9
	1141	2,21	2,24	35,7	13,83	18,9
	1142	2,14	2,31	35,7	13,82	18,9
	1143	2,32	2,11	35,8	13,82	18,9
	1144	2,25	2,15	35,7	13,83	18,9
	1145	2,31	2,17	35,8	13,83	18,9
Syöttö pois	1146	0	0	35,8	13,82	18,9
ennen	1147	0	0	35,6	13,83	18,7
sammuttamista	1148	0	0	35,8	13,83	18,8
	1149	0	0	35,7	13,83	18,9
	1150	0	0	35,8	13,85	18,9
	1151	0	0	35,8	13,86	18,9
	1152	0	0	35,7	13,83	18,9
	1153	0	0	35,7	13,83	18,8
	1154	0	0	35,8	18,85	18,8
	1155	0	0	35,8	13,85	18,9
	Varavoimakone sammutettu:			2018-04-25 09:31:41.586712		

## Liite 11. Prototyypin lopulliset kytkennät

